DISCLAIMER

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof. Reference herein to any social initiative (including but not limited to Diversity, Equity, and Inclusion (DEI); Community Benefits Plans (CBP); Justice 40; etc.) is made by the Author independent of any current requirement by the United States Government and does not constitute or imply endorsement, recommendation, or support by the United States Government or any agency thereof.

Argonne NATIONAL LABORATORY

ANL/NSE-SAS/5.8

The SAS4A/SASSYS-1 Safety Analysis Code System

Version 5.8 (8993)

Nuclear Science and Engineering Division

About Argonne National Laboratory

Argonne National Laboratory is a U.S. Department of Energy laboratory managed by UChicago Argonne, LLC under contract DE-AC02-06CH11357. The Laboratory's main facility is outside Chicago, at 9700 South Cass Avenue, Lemont, Illinois 60439. For information about Argonne and its pioneering science and technology programs, see www.anl.gov.

Document Availability

U.S. Department of Energy (DOE) reports produced after 1991 and a growing number of pre-1991 documents are available free at OSTI.GOV (http://www.osti.gov/), a service of the US Dept. of Energy's Office of Scientific and Technical Information.

Reports not in digital format may be purchased by the public from the National Technical Information Service (NTIS):

U.S. Department of Commerce
National Technical Information Service 5301 Shawnee Rd
Alexandria, VA 22312
www.ntis.gov
Phone: (800) 553-NTIS (6847) or (703) 605-6000
Fax: (703) 605-6900
Email: orders@ntis.gov

Reports not in digital format are available to DOE and DOE contractors from the Office of Scientific and Technical Information (OSTI):

U.S. Department of Energy
Office of Scientific and Technical Information
P.O. Box 62
Oak Ridge, TN 37831-0062
www.osti.gov
Phone: (865) 576-8401
Fax: (865) 576-5728
Email: reports@osti.gov

Disclaimer

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor UChicago Argonne, LLC, nor any of their employees or officers, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of document authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof, Argonne National Laboratory, or UChicago Argonne, LLC.

ANL/NSE-SAS/5.8

The SAS4A/SASSYS-1 Safety Analysis Code System

Version 5.8 (8993)

Nuclear Science and Engineering Division
Argonne National Laboratory

October 09, 2025

Software Copyright

SAS4A/SASSYS-1, Version 5.8
(c) COPYRIGHT UChicago Argonne, LLC - ALL RIGHTS RESERVED

Mini SAS, Version 5.8
(c) COPYRIGHT UChicago Argonne, LLC - ALL RIGHTS RESERVED

THIS SOFTWARE DISCLOSES MATERIAL PROTECTED UNDER COPYRIGHT LAW AND FURTHER DISSEMINATION IS PROHIBITED WITHOUT PRIOR WRITTEN CONSENT OF THE PATENT COUNSEL OF ARGONNE NATIONAL LABORATORY.

Government License

This data was produced by UChicago Argonne, LLC under Contract No. DE-AC02-06CH11357 with the Department of Energy. During the period of commercialization or such other time period specified by DOE, the Government is granted for itself and others acting on its behalf a nonexclusive, paid-up, irrevocable worldwide license in this data to reproduce, prepare derivative works, and perform publicly and display publicly, by or on behalf of the Government. There is provision for the possible extension of the term of this license. Subsequent to that period or any extension granted, the Government is granted for itself and others acting on its behalf a nonexclusive, paid-up, irrevocable worldwide license in this data to reproduce, prepare derivative works, distribute copies to the public, perform publicly and display publicly, and to permit others to do so. The specific term of the license can be identified by inquiry made to UChicago Argonne, LLC or DOE.

THIS SOFTWARE IS SUPPLIED "AS IS" WITHOUT WARRANTY OF ANY KIND.

NEITHER THE UNITED STATES GOVERNMENT, NOR THE UNITED STATES DEPARTMENT OF ENERGY, NOR UCHICAGO ARGONNE, LLC, NOR ANY OF THEIR EMPLOYEES, MAKES ANY WARRANTY, EXPRESS OR IMPLIED, OR ASSUMES ANY LEGAL LIABILITY OR RESPONSIBILITY FOR THE ACCURACY, COMPLETENESS, OR USEFULNESS OF ANY INFORMATION, DATA, APPARATUS, PRODUCT, OR PROCESS DISCLOSED, OR REPRESENTS THAT ITS USE WOULD NOT INFRINGE PRIVATELY OWNED RIGHTS.

Abstract

SAS4A/SASSYS-1 is a software simulation tool used to perform deterministic analysis of anticipated events as well as design basis and beyond design basis accidents for advanced nuclear reactors.

Detailed, mechanistic models of steady-state and transient thermal, hydraulic, kinetic, and mechanical phenomena are employed to describe the response of the reactor core, the reactor primary and secondary coolant loops, the reactor control and protection systems, and the balance-of-plant to accidents caused by changes in coolant flow, loss of heat rejection, or reactivity insertion. The consequences of single and double-fault accidents can be modeled, including fuel and coolant heating, fuel and cladding mechanical behavior, core reactivity feedbacks, coolant loop performance including natural circulation, and decay heat removal. Analyses are typically terminated upon demonstration of reactor and plant shutdown to permanently coolable conditions, or upon violation of design basis margins. The objective of the analysis is to quantify accident consequences as measured by the transient behavior of system performance parameters, such as fuel and cladding temperatures, reactivity, and cladding strain. Originally developed for analysis of sodium cooled reactors with oxide fuel clad by stainless steel, the models were subsequently extended and specialized to metallic fuel clad with advanced alloys and to several other coolant options, including lead, LBE, and water.

For information on the most recent version of the code, please contact sas@anl.gov.

CONTENTS

Part I: Using SAS4A/SASSYS-1 1

1 Introduction 3
- 1.1 SAS4A/SASSYS-1 Background . . . 3
- 1.2 SAS4A/SASSYS-1 Version 5 . . . 5
- 1.3 Documentation Overview . . . 5
- 1.4 References . . . 7

2 SAS4A/SASSYS-1 User's Guide 9
- 2.1 Introduction . . . 9
- 2.2 Modeling Concepts and Code Structure . . . 9
  - 2.2.1 Code Structure Basis . . . 9
  - 2.2.2 Standard Input Data Reading . . . 10
  - 2.2.3 Steady-State Calculation . . . 22
  - 2.2.4 Transient Calculation . . . 22
- 2.3 Data Management . . . 24
- 2.4 Supported Platforms . . . 25
- 2.5 Input and Output Files . . . 25
  - 2.5.1 SAS Input File . . . 25
  - 2.5.2 SAS Output Files . . . 25
  - 2.5.3 Auxiliary Input and Output Files . . . 33
- 2.6 Standard Input Description . . . 34
  - 2.6.1 Input Format . . . 34
  - 2.6.2 Input Data Records Descriptions . . . 36
- 2.7 Input Data Blocks . . . 49
  - 2.7.1 Block 1 — INPCOM — Channel-Independent Options and Integer Input . . . 49
  - 2.7.2 Block 3 — INPMR4 — PRIMAR-4 Integer Input . . . 68
  - 2.7.3 Block 5 — INCONT — Control System Input . . . 93
  - 2.7.4 Block 6 — IINBOP — Balance-of-Plant Integer Input . . . 110
  - 2.7.5 Block 11 — OPCIN — Channel-Independent Floating-Point Input . . . 118
  - 2.7.6 Block 12 — POWINA — Power, Decay-Heat, and Reactivity Input . . . 122
  - 2.7.7 Block 13 — PMATCM — Fuel and Cladding Properties . . . 132
  - 2.7.8 Block 14 — PRIMIN — Primary Loop Input Data . . . 157
  - 2.7.9 Block 15 — FINBOP — Balance-of-Plant Floating-Point Input . . . 159
  - 2.7.10 Block 18 — PMR4IN — PRIMAR-4 Floating-Point Input . . . 171
  - 2.7.11 Block 51 — INPCHN — Channel-Dependent Options and Integer Input . . . 214
  - 2.7.12 Block 61 — GEOMIN — Channel-Dependent Geometry Data . . . 246
  - 2.7.13 Block 62 — POWINC — Channel-Dependent Power and Reactivity Input . . . 250
  - 2.7.14 Block 63 — PMATCH — Channel-Dependent Properties Input . . . 254

2.7.15 Block 64 — COOLIN — Channel-Dependent Coolant Input . . . 261
2.7.16 Block 65 — FUELIN — Channel-Dependent Fuel Input . . . 270
2.8 Example Problems . . . 275
2.8.1 Protected Transient Overpower Example Problem . . . 275
2.8.2 Unprotected Transient Overpower Example Problem . . . 275
2.9 References . . . 278
2.10 Index of Input Variables . . . 278

Part II: Whole-Plant Analysis 315

3 Core Thermal-Hydraulics 317
3.1 Introduction . . . 317
3.2 SAS4A/SASSYS-1 Channel Approach . . . 318
3.2.1 Axial Mesh Structure . . . 321
3.2.2 Radial Mesh Structure . . . 321
3.3 Pre-boiling Transient Heat Transfer, Single Pin Model . . . 325
3.3.1 Core and Axial Blankets . . . 328
3.3.2 Reflector Zones . . . 337
3.3.3 Gas Plenum Region . . . 340
3.3.4 Order of Solution . . . 344
3.3.5 Melting of Fuel or Cladding . . . 345
3.3.6 Coolant Inlet and Re-entry Temperature . . . 345
3.4 Steady-State Thermal Hydraulics . . . 348
3.4.1 Basic Equations . . . 348
3.4.2 Coolant Temperatures . . . 349
3.4.3 Fuel and Cladding Temperatures in the Core and Axial Blankets . . . 349
3.4.4 Structure Temperatures in the Core Axial Blankets . . . 350
3.4.5 Reflector, Structure, Cladding, and Gas Plenum Temperature Outside the Core and Axial Blankets . . . 350
3.5 Transient Heat Transfer after the Start of Boiling . . . 351
3.5.1 Fuel and Cladding Temperatures in the Core and Axial Blanket . . . 351
3.5.2 Structure Temperatures . . . 352
3.5.3 Reflector Temperatures . . . 355
3.5.4 Gas Plenum Region . . . 357
3.5.5 Coolant Temperatures in Liquid Slugs . . . 358
3.6 Fuel-Cladding Bond Gap Conductance . . . 362
3.7 Fuel Pin Heat-transfer After Pin Disruption or Relocation of Fuel or Cladding . . . 363
3.7.1 Fuel-pin Heat Transfer After Pin Disruption in PLUTO2 or LEVITATE . . . 363
3.8 Heat-transfer Time Step Control . . . 364
3.9 Steady-State and Single-Phase Transient Hydraulics . . . 366
3.9.1 Introduction . . . 366
3.9.2 Basic Equations . . . 369
3.9.3 Flow Orifices . . . 371
3.9.4 Finite Difference Equations - Coolant Flow Rates . . . 373
3.9.5 Coolant Pressures . . . 374
3.10 Multiple-Pin Model . . . 375
3.10.1 Introduction . . . 375
3.10.2 Physical Model . . . 376
3.10.3 Numerical Methods . . . 379
3.10.4 Detailed Mathematical Treatment . . . 380
3.10.5 Relationship Between Single Pin and Multiple Pin Models . . . 388
3.11 Subassembly-to-subassembly Heat Transfer . . . 388

3.12 Interaction with Other Models . . . 389
3.12.1 Reactivity Feedback . . . 389
3.12.2 Coupling Between Core Channels and PRIMAR-4 . . . 389
3.13 Subroutine Descriptions and Flowcharts . . . 391
3.14 Subchannel Treatment Details . . . 393
3.14.1 Model Features . . . 396
3.14.2 Basic Equations . . . 397
3.14.3 Numerical Procedures . . . 401
3.14.4 Interfacing with Other Models in the Code . . . 401
3.14.5 Data Management . . . 404
3.14.6 Coding Overview and Subroutines . . . 404
3.14.7 Required Computer Capabilities . . . 408
3.15 Input and Output . . . 410
3.15.1 Input Description . . . 410
3.15.2 Output Description . . . 414
3.15.3 Subchannel Information . . . 418
3.16 Thermal Properties of Fuel and Cladding . . . 425
3.16.1 Fuel Density . . . 426
3.16.2 Fuel Thermal Conductivity . . . 426
3.17 References . . . 428
3.18 Nomenclature . . . 429
3.19 Appendices . . . 433
3.19.1 APPENDIX 3.1: Degree of Implicitness for Flow and Temperature Calculations . . . 433
3.19.2 APPENDIX 3.2: Input Preprocessor . . . 438
3.19.3 APPENDIX 3.3: Visualization . . . 457
3.19.4 APPENDIX 3.4: Coolant Physical Properties . . . 470

4 Reactor Point Kinetics, Decay Heat, and Reactivity Feedback 493
4.1 Introduction . . . 493
4.2 Reactor Power . . . 493
4.3 Delayed-Neutron Precursors . . . 494
4.4 Decay Heat . . . 495
4.4.1 The ANS Decay Heat Power Standard . . . 495
4.4.2 Implementation . . . 497
4.5 Net Reactivity . . . 499
4.5.1 User-Programmed Reactivity . . . 500
4.5.2 Control System Reactivity . . . 500
4.5.3 Fuel Doppler Feedback Reactivity . . . 500
4.5.4 Fuel, Cladding, and Structure Axial Expansion Feedback Reactivity . . . 501
4.5.5 Coolant Density Feedback Reactivity . . . 507
4.5.6 Radial Expansion Feedback Reactivity . . . 508
4.5.7 Control Rod Drive Expansion Feedback Reactivity . . . 516
4.5.8 Fuel and Cladding Relocation Feedback Reactivity . . . 517
4.5.9 EBR-II Reactivity Feedback model . . . 518
4.6 Solution Methods . . . 525
4.6.1 Point Kinetics . . . 525
4.6.2 Specified Power History . . . 527
4.7 Code Organization and Data Flow . . . 527
4.8 Input/Output Description . . . 528
4.9 References . . . 532
4.10 Nomenclature . . . 533
4.11 Appendices . . . 537
4.11.1 Appendix 4.1: Steady-State Power Normalization in SAS4A/SASSYS-1 . . . 537

5 Primary and Intermediate Loop Thermal Hydraulics Module **539**
5.1 Introduction . . . 539
5.2 Hydraulic Calculations . . . 541
5.2.1 Compressible Volumes and Liquid Segments . . . 541
5.2.2 Estimated Core Flow . . . 546
5.2.3 Method of Solution . . . 547
5.2.4 Degree of Implicitness . . . 548
5.3 Specific Components: Contributions to Flow and Pressure Equations . . . 549
5.3.1 Compressible Volumes Without Cover Gas . . . 549
5.3.2 Compressible Volumes With Cover Gas . . . 551
5.3.3 Pipes and Intermediate Heat Exchangers . . . 555
5.3.4 Pumps . . . 560
5.3.5 Valves . . . 587
5.3.6 Check Valves . . . 587
5.4 Liquid Temperature Calculations . . . 587
5.4.1 Pipe Temperatures . . . 588
5.4.2 Heat Exchangers: Detailed Options . . . 596
5.4.3 Bypass Channels . . . 604
5.4.4 Compressible Volumes . . . 610
5.4.5 Table Look-Up for Steam Generators or Intermediate Heat Exchangers . . . 614
5.4.6 Component-to-Component Heat Transfer . . . 616
5.4.7 RVACS/RACS Models . . . 619
5.4.8 Stratified Volume Model . . . 641
5.5 Steam Generator . . . 646
5.6 Air Dump Heat Exchanger . . . 646
5.6.1 Methodology . . . 646
5.6.2 Steady-State Solution . . . 650
5.7 Cover Gas Flow and Pressure Calculations . . . 650
5.7.1 Introduction . . . 650
5.7.2 Basic Equations . . . 651
5.7.3 Solution . . . 653
5.8 Overall Solution and Time Step Control . . . 654
5.8.1 Order of Calculation in PRIMAR . . . 654
5.8.2 PRIMAR Time Step Size . . . 655
5.9 Steady-State Initialization . . . 655
5.9.1 Core Channels, Inlet Plenum, and Outlet Plenum . . . 656
5.9.2 Primary Heat Transport System . . . 657
5.9.3 Intermediate Heat Transport System . . . 658
5.9.4 Individual Components . . . 659
5.9.5 Null Transient for Steady-State Initialization . . . 661
5.10 PRIMAR-1 Option . . . 663
5.10.1 Defined Pump Head . . . 663
5.10.2 Defined Flow Rate . . . 664
5.11 Interaction With Other Models . . . 665
5.11.1 Information Received by PRIMAR from Other Models . . . 665
5.11.2 Information Supplied by PRIMAR to Other Models . . . 666
5.12 Subroutine Descriptions and Flow Charts . . . 667
5.12.1 Subroutine Listing and Description . . . 667
5.12.2 PRIMAR-4 Module Flow Chart . . . 667
5.13 Input/Output Description . . . 670
5.13.1 Input Description . . . 670
5.13.2 Example of a Primary Loop . . . 675

5.13.3 Output Description . . . 677
5.13.4 Plotting PRIMAR-4 Data . . . 682
5.14 References . . . 688
5.15 Nomenclature . . . 688
5.16 Appendices . . . 694
5.16.1 Appendix 5.1: IHX Matrix Coefficients . . . 694
5.16.2 Appendix 5.2: IHX Matrix Solution Algorithm . . . 697
5.16.3 Appendix 5.3: Cover Gas Flow and Pressure Algorithm . . . 699
5.16.4 Appendix 5.4: Air Blast Heat Exchanger Stack Momentum Equation . . . 702
5.16.5 Appendix 5.5: Interpolation from User-Supplied Tables . . . 703
5.16.6 Appendix 5.6: Optional Eulerian Solution for Pipe Temperatures . . . 706
5.16.7 Appendix 5.7: Detailed Derivation of EQ EM Pump . . . 707

6 Control System **713**
6.1 Introduction . . . 713
6.2 Generalized Model . . . 714
6.2.1 General Equations . . . 714
6.2.2 Block Diagram Approach . . . 715
6.3 Solution Techniques . . . 715
6.3.1 Definition of Block Diagram . . . 716
6.3.2 Steady-State Solution . . . 717
6.3.3 Transient Solution . . . 717
6.4 Guide to User Application . . . 719
6.4.1 Signal Definition Rules . . . 719
6.4.2 Units . . . 720
6.4.3 Initial Conditions . . . 720
6.4.4 Solution Accuracy . . . 720
6.4.5 Solution Stability . . . 721
6.4.6 Zero Crossing Parameter . . . 721
6.5 Control System Signals . . . 722
6.5.1 Measured Signals . . . 722
6.5.2 Demand Signals . . . 722
6.5.3 Block Signals . . . 722
6.5.4 Control Signals . . . 726
6.6 References . . . 727
6.7 Nomenclature . . . 727
6.8 Appendix 6.1: Control System Input Description . . . 727
6.8.1 Input Block Structure . . . 727
6.8.2 Block Identifier Card . . . 728
6.8.3 Signal Cards . . . 728
6.8.4 Data Cards . . . 739
6.8.5 Sample Input . . . 740

7 Balance of Plant **743**
7.1 Introduction . . . 743
7.2 Network Thermal/Hydraulic Model . . . 744
7.2.1 Discretization of the Balance of Plant . . . 744
7.2.2 Analytical Equations . . . 744
7.2.3 Discretized Equations . . . 746
7.2.4 Solution Procedure . . . 755
7.2.5 Coupling Between the Balance-of-Plant Model and the Steam Generator Model . . . 756
7.2.6 Implementation of the Balance-of-Plant Model in SASSYS-1 . . . 757
7.2.7 Creating a Plant Nodalization . . . 765

7.3 Steam Generator Model . . . 769
7.3.1 Once-Through Steam Generator . . . 769
7.3.2 Recirculation-Type Steam Generator . . . 773
7.3.3 Numerical Solution Methods . . . 775
7.4 Component Models . . . 793
7.4.1 General Assumptions . . . 793
7.4.2 Deaerator . . . 794
7.4.3 Steam Drum . . . 798
7.4.4 Condenser . . . 799
7.4.5 Reheater . . . 802
7.4.6 Flashed Heater . . . 806
7.4.7 Drain Cooler . . . 807
7.4.8 Desuperheating Heater . . . 809
7.4.9 Desuperheater/Drain Cooler . . . 811
7.4.10 Turbine . . . 811
7.4.11 Relief Valve . . . 821
7.4.12 Steady-State Initialization . . . 823
7.4.13 Code Implementation and Operation . . . 824
7.5 References . . . 826
7.6 Nomenclature . . . 827
7.7 Appendices . . . 830
7.7.1 Listing of Balance-of-Plant Variables . . . 830
7.7.2 Steam Generator Water-Side Heat Transfer Correlations . . . 844
7.7.3 Two-Phase Interface Solution Scheme for Heater Cylinders Lying on the Side . . . 845
7.7.4 Dictionary of Steam Generator Model Variables . . . 847
7.7.5 Material Properties Data . . . 855

Part III: Fuel Performance **859**

8 DEFORM-4: Steady-State and Transient Pre-Failure Pin Behavior **861**
8.1 Overview and General Background . . . 861
8.1.1 Objectives, General Background, and General Physical Description . . . 861
8.1.2 Interaction with other SAS4A Models . . . 866
8.2 Fuel-pin Mechanics . . . 868
8.2.1 Solid Fuel and Cladding Thermoelastic Solution . . . 870
8.2.2 Cracked Fuel Thermoelastic Solution and Crack Volume . . . 874
8.2.3 Fully Cracked Fuel . . . 876
8.2.4 Axial Expansion Solution . . . 877
8.3 Fuel-pin Phenomenology . . . 879
8.3.1 As-Fabricated Porosity Migration . . . 880
8.3.2 Grain Growth . . . 883
8.3.3 Fission-gas Release . . . 886
8.3.4 Fuel Swelling . . . 891
8.3.5 Irradiation-induced Cladding Swelling . . . 897
8.3.6 Fission-gas Plenum Pressure . . . 899
8.3.7 Molten Cavity Pressurization . . . 901
8.3.8 Fuel-cladding Gap Conductance . . . 910
8.3.9 Fuel Axial Expansion Reactivity Model . . . 917
8.3.10 Metal Fuel Modeling . . . 918
8.4 Fuel-pin Failure Criteria . . . 925
8.4.1 Input Specified Failure Criteria . . . 925
8.4.2 Calculated Failure Correlations . . . 926

8.4.3 Preliminary Metal Fuel Failure Criteria . . . 929
8.4.4 Failure Modeling Coupling to Fuel Motion Models . . . 930
8.4.5 Time-step Control on Approach to Failure . . . 932
8.5 General Method of Solution . . . 932
8.5.1 Thermal-mechanical Solution . . . 937
8.5.2 Plastic Fuel Deformation Solution . . . 940
8.5.3 Final DEFORM-4 Calculations . . . 942
8.6 Computer Code Implementation . . . 942
8.6.1 Subroutine and Function List . . . 942
8.6.2 Input Variables . . . 944
8.6.3 Output Specific to DEFORM-4 . . . 949
8.7 Material Properties . . . 964
8.7.1 Fuel Density . . . 964
8.7.2 Fuel Thermal Expansion Coefficient . . . 965
8.7.3 Fuel Modulus of Elasticity . . . 966
8.7.4 Mixed Oxide Fuel Thermal Conductivity . . . 966
8.7.5 Mixed Oxide Fuel Fracture Strength . . . 968
8.7.6 Mixed Oxide Fuel Creep Rate . . . 968
8.7.7 Cladding Thermal Expansion Coefficient . . . 969
8.7.8 Cladding Modulus of Elasticity . . . 969
8.7.9 Cladding Ultimate Tensile Strength . . . 969
8.7.10 Cladding Flow Stress . . . 970
8.7.11 Thermal Conductivity of Helium . . . 973
8.7.12 Thermal Conductivity of Fission Gas . . . 973
8.7.13 Fuel Hardness . . . 973
8.7.14 Cladding Hardness . . . 974
8.7.15 Cladding Fast Creep . . . 974
8.7.16 Material Properties of Metal Alloy Fuel . . . 974
8.7.17 15-15Ti Cladding Properties . . . 976
8.8 References . . . 980
8.9 Nomenclature . . . 982

9 MFUEL: Metallic Fuel Performance Model 985
9.1 Introduction . . . 985
9.2 Model Formulation . . . 985
9.2.1 Fuel Constituent Redistribution . . . 985
9.2.2 Fuel Clad Chemical Interaction . . . 988
9.2.3 Fuel Swelling and Fission Gas Release . . . 994
9.2.4 Plenum Gas Pressure . . . 996
9.2.5 Fuel Anisotropic Axial Elongation . . . 997
9.2.6 Fuel Pin Mechanical Analysis . . . 999
9.2.7 In-Pin Sodium Treatment . . . 1008
9.2.8 Clad Outer Corrosion . . . 1009
9.2.9 CDF-Based Clad Creep Rupture . . . 1009
9.2.10 Mechanistic Clad Creep Rupture . . . 1012
9.2.11 Clad Failure Assessment . . . 1015
9.3 Time Step Controls . . . 1016
9.4 Solution Algorithm and Flow Charts . . . 1016
9.5 Interactions with Other SAS Models . . . 1018
9.6 Input and Output Description . . . 1022
9.7 References . . . 1024
9.8 Appendices . . . 1025
9.8.1 Appendix 9.1: Main Assumptions . . . 1025

9.8.2 Appendix 9.2: Thermodynamics and Kinetics Parameters of Fuel Constituent Redistribution Model . . . 1027
9.8.3 Appendix 9.3: Thermodynamics and Kinetics Parameters of Fuel Clad Chemical Interaction Model . . . 1028
9.8.4 Appendix 9.4: Fuel Swelling and Fission Gas Release Model Parameters . . . 1030
9.8.5 Appendix 9.5: Mechanical Properties . . . 1032
9.8.6 Appendix 9.6: Metallic Fuel Thermal Conductivity Model . . . 1034
9.8.7 Appendix 9.7: MCF Model Parameters . . . 1034
9.8.8 Appendix 9.8: DEFORM-5: Metallic Fuel Cladding Transient Behavior Model . . . 1035

10 SSCOMP: Pre-Transient Characterization of Metallic Fuel Pins 1045
10.1 Introduction and Overview . . . 1045
10.2 The Earlier SSCOMP Model and Current Capability . . . 1046
10.3 Thermal Properties of U-Pu-Zr and U-Zr Alloy Fuels . . . 1047
10.3.1 Temperature-Dependent Tables . . . 1054
10.3.2 Regionwise Interpolation of IFR Handbook Data . . . 1054
10.3.3 Correlations of the Report ANL/RAS 85-19 . . . 1151
10.3.4 Properties of Some Specific Fuels . . . 1167
10.3.5 Computer Implementation . . . 1172
10.4 Future Directions for Modeling Efforts . . . 1176
10.4.1 Fuel constituent Radial Migration . . . 1177
10.4.2 Fission Gas Behavior, and Porosity Formation and Distribution . . . 1179
10.4.3 Irradiation-Induced Radial and Axial Swelling of Fuel and Cladding . . . 1179
10.5 References . . . 1179
10.6 Nomenclature . . . 1180
10.7 Appendices . . . 1182
10.7.1 APPENDIX 10.1: Equivalency of Interpolation in Terms of Weight Fractions and Atom Fractions . . . 1182
10.7.2 APPENDIX 10.2: Correlations in the IFR Handbook and Some Other Sources . . . 1184

11 FPIN2: Pre-Failure Metal Fuel Pin Behavior Model 1187
11.1 Introduction . . . 1187
11.1.1 Brief Description of FPIN2 . . . 1187
11.1.2 Overview . . . 1188
11.2 Fuel Element Mechanics . . . 1188
11.2.1 Basic Stress-Strain-Displacement Equations . . . 1189
11.2.2 Algorithm for Separating the Components of the Total Strain . . . 1195
11.2.3 Continuous Cracking Model . . . 1197
11.2.4 Finite Element Formulation . . . 1198
11.2.5 Large Strain Analysis . . . 1202
11.3 Metal Fuel Element Models . . . 1204
11.3.1 Central Cavity and Plenum Models . . . 1204
11.3.2 Extrusion of Molten Fuel into the Plenum . . . 1207
11.3.3 Basic Metal Fuel Properties . . . 1207
11.3.4 Secondary Creep of Metallic Fuels . . . 1208
11.3.5 Fission Induced Gas Swelling of Metal Fuels . . . 1210
11.3.6 Generalized Plastic Flow Behavior of Cladding Materials . . . 1212
11.3.7 Fuel-Cladding Eutectic Formation . . . 1215
11.3.8 Fuel Element Failure . . . 1216
11.4 Integration into SAS4A/SASSYS-1 and Usage of Integrated Model . . . 1216
11.4.1 SAS-FPIN2 Coupling Methodology . . . 1217
11.4.2 Input Description for Interfaced SAS-FPIN2 Calculation . . . 1221
11.4.3 Pre-transient Fuel Element Characterization . . . 1228

11.4.4 Output Description and Graphics File Usage . . . 1231
11.5 References . . . 1233
11.6 Nomenclature . . . 1234
11.7 Appendices . . . 1237
11.7.1 Explicit Formulas for Stiffness Matrix and Load Vector . . . 1237
11.7.2 List of Input Variables for Standalone FPIN2 Calculation . . . 1238

Part IV: Failure Modeling **1245**

12 Coolant Voiding **1247**
12.1 Introduction . . . 1247
12.2 Liquid Slug Flow Rates . . . 1249
12.3 Interface Velocities . . . 1256
12.4 Bubble Formation and Collapse . . . 1256
12.5 Uniform Vapor Pressure Model: Small Vapor Bubbles . . . 1259
12.5.1 Heat Flow to Vapor from Cladding and Structure . . . 1260
12.5.2 Heat Flow through Liquid-Vapor Interface . . . 1267
12.5.3 Change in Vapor Energy . . . 1273
12.5.4 Energy Balance . . . 1275
12.5.5 Vapor Temperatures . . . 1276
12.6 Vapor Pressure Gradient Model: Large Vapor Bubbles . . . 1277
12.6.1 Continuity and Momentum Equations . . . 1277
12.6.2 Finite Differencing . . . 1280
12.6.3 Simultaneous Solution of the Differenced, Linearized Mass and Momentum Equations 1303
12.7 Voiding Due to Gas Release from Failed Fuel Pins . . . 1305
12.8 Time Step Controls . . . 1309
12.9 Interaction with Other Models . . . 1310
12.10 Detailed Flow Description . . . 1311
12.11 Input Description . . . 1314
12.12 Sample Output . . . 1328
12.13 Physical Properties of Sodium . . . 1330
12.14 References . . . 1336
12.15 Nomenclature . . . 1338
12.16 Appendices . . . 1340
12.16.1 Derivation of the Expression for the Spatial Derivative of the Liquid Temperature at a Liquid-Vapor Interface . . . 1340
12.16.2 Gaussian Elimination Solution of Linearized Vapor-Pressure-Gradient Equations . . . 1343

13 CLAP: Cladding Motion Model **1347**
13.1 Introduction and Overview . . . 1347
13.1.1 Background and Description of the CLAP Physical Model . . . 1347
13.1.2 CLAP Module Structure and Interaction with Other SAS4A Models . . . 1348
13.2 Mathematical Model . . . 1348
13.2.1 Sodium Vapor Flow Model . . . 1348
13.2.2 Moving Cladding Basic Equations . . . 1352
13.2.3 Refrozen Cladding Basic Equations . . . 1354
13.2.4 Heat-transfer Relationships . . . 1354
13.2.5 Cladding Physical Properties Relationships . . . 1357
13.3 Method of Solution . . . 1357
13.3.1 CLAP Initialization and Cladding Meltthrough . . . 1357
13.3.2 Solution of the Cladding Energy Equations . . . 1358
13.3.3 Refrozen Steel Area Calculation . . . 1365

13.3.4 Moving Cladding Area Calculation . . . 1365
13.3.5 Reactivity Calculation . . . 1366
13.3.6 Moving Cladding Velocity Calculation . . . 1367
13.4 Input-Output Description . . . 1369
13.5 References . . . 1370
13.6 Nomenclature . . . 1371

14 PLUTO2: Non-Voided Channel Fuel Motion Analysis 1375
14.1 Introduction and Overview . . . 1375
14.1.1 Historical Background and Description of the Physical Model . . . 1375
14.1.2 Overview of the Program Flow . . . 1382
14.1.3 Interaction with other SAS4A Modules . . . 1389
14.2 In-Pin Fuel Motion . . . 1389
14.2.1 Overview and Assumptions . . . 1389
14.2.2 Initial Conditions for the In-pin Calculation from DEFORM and the Pre-Failure Pin Heat-transfer Calculation . . . 1392
14.2.3 Coupling of the In-pin Motion Calculation with the PLHTR Heat Transfer Calculation 1392
14.2.4 Overview of the Numerical Approach for the In-pin Fuel Motion Calculation and Description of Subroutines PLlPIN and PL2PIN . . . 1393
14.2.5 Definition of the Generalized Smear Densities for the In-pin Calculation . . . 1394
14.2.6 Differential Equations for the In-pin Fuel Motion and Description of Sink and Source Terms . . . 1395
14.2.7 Finite Difference Equations for the In-pin Motion . . . 1403
14.2.8 Time-step Determination for the In-Pin Motion . . . 1406
14.3 Fuel and Fission-gas Ejection from the Pins . . . 1407
14.3.1 Physical Model and Assumptions . . . 1407
14.3.2 Numerical Solution of the Fuel and Gas Ejection Calculation . . . 1408
14.3.3 Axial Pin Failure Propagation . . . 1409
14.3.4 Complete Pin Disruption and Switch to LEVITATE . . . 1411
14.4 Channel Hydrodynamics Model . . . 1412
14.4.1 Overview and Definition of Generalized Smear Densities . . . 1412
14.4.2 Mass Conservation for Fuel, Sodium, and Free and Dissolved Fission-gas . . . 1419
14.4.3 Fuel Flow Regimes, Fuel Plateout and Frozen Crust Release, Plated-out and Moving Fuel Configurations, and Energy and Momentum Exchange Terms . . . 1426
14.4.4 Mobile Fuel and Fuel Crust Energy Equations . . . 1450
14.4.5 Sodium/Fission-gas Energy Equation and Channel Pressure Calculation . . . 1454
14.4.6 Momentum Equations in the Coolant Channel . . . 1456
14.5 Temperature Calculation of Cladding, Structure, Reflector, and Liquid Sodium . . . 1467
14.5.1 Liquid Sodium, Cladding, Structure, and Reflector Temperature Calculation Outside of the Interaction Region . . . 1467
14.5.2 Cladding and Reflector Temperature Calculation in the Interaction Region . . . 1469
14.5.3 Structure Temperature Calculation in the Interaction Region . . . 1471
14.6 Interaction with the Point Kinetics and the Primary Loop Module . . . 1471
14.6.1 Interaction With the Point Kinetics Module . . . 1471
14.6.2 Coupling with the Primary Loop Module . . . 1472
14.7 Code Logic Description . . . 1473
14.7.1 PLUTO2 Initialization . . . 1473
14.7.2 Time Step Considerations and Auxiliary Subroutines . . . 1474
14.8 Description of Input to and Output of the PLUTO2 Module . . . 1477
14.8.1 PLUTO2 Input . . . 1477
14.8.2 PLUTO2 Output . . . 1479
14.9 References . . . 1496
14.10 Nomenclature . . . 1502

15 PINACLE: In-Pin Pre-Failure Molten Fuel Relocation Module **1511**
15.1 Overview . . . 1511
15.1.1 Historical Background . . . 1511
15.1.2 Physical Description of the PINACLE Model . . . 1514
15.1.3 Interaction of PINACLE with other Models Within the SAS4A System . . . 1515
15.2 In-pin Hydrodynamic Model . . . 1520
15.2.1 Overview of the Numerical Approach for the In-pin Fuel Motion Calculation and Description of Subroutines PN1PIN and PN2PIN . . . 1520
15.2.2 Definition of the Generalized Smear Densities for the In-pin Hydrodynamic Calculation . . . 1521
15.2.3 Differential Equations for the In-Pin Fuel Motion and Description of Sink and Source Terms . . . 1523
15.2.4 Finite Difference Equations for the In-Pin Motion Model . . . 1529
15.2.5 Treatment of the Fuel Ejection Above the Top of the Active Fuel Pin . . . 1532
15.2.6 Time-step Determination for the Pre-Failure In-Pin Motion . . . 1536
15.3 Computer Implementation . . . 1537
15.3.1 Detailed Logic Flow Description . . . 1537
15.3.2 Input Parameters Relevant to PINACLE . . . 1539
15.3.3 Output Description . . . 1541
15.4 Future Directions for Modeling Efforts. . . . 1544
15.4.1 Annular Molten Region . . . 1544
15.4.2 New Moving Material Components . . . 1545
15.4.3 Inner Cladding Ablation . . . 1547
15.4.4 Composition-Dependent Moving Mixture Properties . . . 1547
15.4.5 Fuel Freezing in the Plenum . . . 1547
15.4.6 Initiation of Axial Motion . . . 1547
15.4.7 Fuel Blanket and Sodium Slug Motion . . . 1548
15.4.8 PINACLE Termination Upon Fuel Freezing and Restart Upon Remelting . . . 1548
15.5 References . . . 1549
15.6 Nomenclature . . . 1549

16 LEVITATE: Voided Fuel Motion **1553**
16.1 Overview . . . 1553
16.1.1 Historical Background and Physical Description of the LEVITATE Model . . . 1553
16.1.2 Physical Description of the LEVITATE Model . . . 1554
16.1.3 Interaction of LEVITATE with Other Models within the SAS4A System . . . 1558
16.1.4 Improvements and New Models Relevant to the LEVITATE Module in SAS4A Version 1.1 . . . 1564
16.2 In-pin Hydrodynamic Model . . . 1565
16.2.1 Physical Models . . . 1565
16.2.2 Method of Solution and General Numerical Considerations . . . 1566
16.2.3 Finite Difference Forms and Solution Technique, Special Situations . . . 1567
16.3 Fuel Ejection from the Pins and Fuel-pin Disruption . . . 1568
16.3.1 Fuel Ejection via a Cladding Rupture . . . 1568
16.3.2 Fuel-pin Disruption . . . 1573
16.3.3 Fuel Ejection via the Open Pin Stubs . . . 1574
16.4 Coolant Channel Hydrodynamic Model . . . 1576
16.4.1 Physical Models . . . 1576
16.4.2 Method of Solution and general Numerical Considerations . . . 1578
16.4.3 Differential Equations and Finite Difference Equations . . . 1590
16.5 Freezing, Melting and Heat-Transfer Processes . . . 1642
16.5.1 Physical Models . . . 1642
16.5.2 Description of the Method of Solution and Logic Flow . . . 1643

16.5.3 Fuel/steel Freezing and Crust Formation . . . 1643
16.5.4 Fuel Crust Breakup and Remelting . . . 1647
16.5.5 Cladding and Hexcan Ablation . . . 1648
16.5.6 Fuel/Steel Chunk Melting . . . 1649
16.5.7 Heat-transfer Calculation for the Steel Cladding and the Hexcan Wall . . . 1650
16.6 LEVITATE Interaction with Other Modules . . . 1653
16.6.1 Interaction with the Point Kinetics Module . . . 1653
16.6.2 Coupling with the Primary Loop Module . . . 1654
16.7 Detailed Logic Flow Description . . . 1655
16.7.1 Data Management Considerations . . . 1655
16.7.2 Logic Flow for Solution Advancement . . . 1655
16.7.3 Time-step Considerations . . . 1658
16.7.4 List of LEVITATE Subroutines . . . 1659
16.8 Input Parameters Relevant to LEVITATE . . . 1660
16.9 Output Description . . . 1666
16.9.1 Regular Output . . . 1666
16.9.2 Auxiliary and Debug Output . . . 1672
16.10 References . . . 1677
16.11 Nomenclature . . . 1678

Part I: Using SAS4A/SASSYS-1

CHAPTER

ONE

INTRODUCTION

The SAS4A/SASSYS-1 computer code is developed by Argonne National Laboratory for thermal, hydraulic, and neutronic analysis of power and flow transients in liquid-metal-cooled nuclear reactors (LMRs). SAS4A was developed to analyze severe core disruption accidents with coolant boiling and fuel melting and relocation, initiated by a very low probability coincidence of an accident precursor and failure of one or more safety systems. SASSYS-1, originally developed to address loss-of-decay-heat-removal accidents, has evolved into a tool for margin assessment in design basis accident (DBA) analysis and for consequence assessment in beyond-design-basis accident (BDBA) analysis.

SAS4A contains detailed, mechanistic models of transient thermal, hydraulic, neutronic, and mechanical phenomena to describe the response of the reactor core, its coolant, fuel elements, and structural members to accident conditions. The core channel models in SAS4A provide the capability to analyze the initial phase of core disruptive accidents, through coolant heat-up and boiling, fuel element failure, and fuel melting and relocation. Originally developed to analyze oxide fuel clad with stainless steel, the models in SAS4A have been extended and specialized to metallic fuel with advanced alloy cladding.

SASSYS-1 provides the capability to perform a detailed thermal/hydraulic simulation of the primary and secondary sodium coolant circuits and the balance-of-plant steam/water circuit. These sodium and steam circuit models include component models for heat exchangers, pumps, valves, turbines, and condensers, and thermal/hydraulic models of pipes and plena. SASSYS-1 also contains a plant protection and control system modeling capability, which provides digital representations of reactor, pump, and valve controllers and their response to input signal changes.

1.1 SAS4A/SASSYS-1 Background

In the late 1960s, the then U.S. Atomic Energy Commission gave development of a liquid-metal-cooled fast reactor (LMR) a high priority, and the development of the Fast Flux Test Facility (FFTF) became a cornerstone of that program. To provide adequate support for the FFTF and for the expected LMRs to follow, a major base technology program was established which provided a continuous stream of experimental information and design correlations. This experimental data would either confirm design choices or prove the need for design modifications. At the time, the “tremendous amount of data and experience pertaining to thermal design” of LMRs was recognized as providing the technical foundation for the future commercial development of LMRs.[1-1]

Along with the generation of experimental data came the development of safety analysis methods that used that data in correlations for mechanistic, probabilistic, or phenomenological models. These models were developed for a variety of needs ranging from individual components, such as heat exchangers, pumps, or containment barriers, to whole core or even whole-plant dynamics. A major portion of the overall technical effort since that time has been allocated to safety considerations, and the SAS4A/SASSYS-1 safety analysis code is the result of that dedication.

Perhaps the strongest factor that influenced early fast reactor safety analysis was the concern over the possibility of core compaction followed by an energetic core disassembly — the so-called Bethe-Tait accident.[1-2] In the late 1960s, the Hanford Engineering Development Laboratory (HEDL) began developing the MELT code[1-3,1-4] to evaluate the initiating phase of hypothetical core disruption accidents (HCDA) as part of the FFTF project. The MELT series of codes has the capability to model the transient behavior of several representative fuel pins (channels) within a reactor core to allow for incoherency in the accident sequence. By 1978 MELT had evolved into the MELT-IIIB code.[1-4]

Around the same time that development on MELT began, Argonne National Laboratory began developing the SAS series of codes.[1-5-1-9] Like MELT, SAS has the capability to model the transient behavior of several representative channels to evaluate the initiating phase of HCDAs. SAS1A originated from a sodium boiling model and includes single- and two-phase coolant flow dynamics, fuel and cladding thermal expansion and deformation, molten fuel dynamics, and a point kinetics model with reactivity feedback. By 1974, SAS evolved to the SAS2A computer code[1-6] which included a detailed multiple slug and bubble coolant boiling model which greatly enhanced the ability to simulate the initiating phases of loss-of-flow (LOF) and transient overpower (TOP) accidents up to the point of cladding failure and fuel and cladding melting.

The SAS3A code [1-7] added mechanistic models of fuel and cladding melting and relocation. This version of the code was used extensively for analysis of accidents in the licensing of FFTF. In anticipation of LOF and TOP analysis requirements for licensing of the Clinch River Breeder Reactor Plant (CRBRP), new fuel element deformation, disruption, and material relocation models were written for the SAS4A version of the code,[1-8] which saw extensive validation against TREAT M-Series test data. In addition, a variant of SAS4A, named SASSYS-1, was developed with the capability to model ex-reactor coolant systems to permit the analysis of accident sequences involving or initiated by loss of heat removal or other coolant system events. This allows the simulation of whole-plant dynamics feedback for both shutdown and off-normal conditions, which have been validated against EBR-II Shutdown Heat Removal Test (SHRT) data and data from the FFTF LOF tests.

Although SAS4A and SASSYS-1 are generally portrayed as two computer codes, they have always shared a common code architecture, the same data management strategy, and the same core channel representation. Subsequently, the two code branches were merged into a single code referred to as SAS4A/SASSYS-1. Version 2.1 of the SAS4A/SASSYS-1 code [1-10,1-11] was distributed to Germany, France, and Japan in the late 1980s, and it serves as a common tool for international oxide fuel model developments.

Beyond the release of SAS4A/SASSYS-1 v 2.1, revisions to SAS4A/SASSYS-1 continued throughout the Integral Fast Reactor (IFR) program between 1984 and 1994,[1-12] culminating with the completion of SAS4A/SASSYS-1 v 3.0 in 1994.[1-13] During this time, the modeling emphasis shifted towards metallic fuel and accident prevention by means of inherent safety mechanisms. This resulted in 1) addition of new models and modification of existing models to treat metallic fuel, its properties, behavior, and accident phenomena, and 2) addition and validation of new capabilities for calculating whole-plant design basis transients, with emphasis on the EBR-II reactor and plant [1-14], the IFR prototype. The whole-plant dynamics capability of the SASSYS-1 component plays a vital role in predicting passive safety feedback. Without it, meaningful boundary conditions for the core channel models are not available, and accident progression is not reliably predicted.

By the mid 1990s, SAS4A/SASSYS-1 v 3.1 had been completed as a significant maintenance update, but it was not released until 2012.[1-15]

1.2 SAS4A/SASSYS-1 Version 5

In the time since the development of Version 3, several modeling additions and enhancements have been made to meet U.S. Department of Energy programmatic needs. Significant among these are

- Detailed sub-channel models for whole-core analyses to resolve intra-assembly temperature and flow distributions [1-9]
- 3D visualization capabilities for sub-channel results.
- Extended decay-heat models to support long-term transients and complex, actinide-bearing fuels.
- Support for coupling with external CFD simulations to resolve flow distribution and thermal stratification effects.
- Treatment of axial expansion feedback from assembly duct walls
- Support for spatial kinetics (requires DIF3D-K)
- Extension of the control-system model to include sinusoidal functions that can be used to represent seismic oscillation effects.
- Addition for heavy liquid-metal coolants (lead and lead-bismuth eutectic)
- Support for user-defined coolant properties
- Detailed steam-generator model updates
- New electromagnetic pump modeling capabilities
- Several bug fixes and other enhancements

In addition to the above, a major restructuring of the code has been completed to adapt all source files to free-form source format and new model developments are being implemented using modern object-oriented practices.

1.3 Documentation Overview

The rest of this manual contains details of the modeling capabilities of SAS4A/SASSYS-1. The chapter organization shown in Table 1.3.1 reflects the major model delineations. Each chapter provides in-depth descriptions of the models, including model formulations, solution techniques, and input descriptions. It is critical that users understand the relationships between their model input and the model formulations given in this manual. Failure to understand these relationships can result in broken models and misleading results.

SAS4A/SASSYS-1 provides a detailed, multiple-channel thermal/hydraulic treatment of the reactor core. Each channel represents a fuel pin, its cladding, the associated coolant, and a fraction of the subassembly duct wall. Other positioning hardware, such as wire wraps or grid spacers, is usually lumped into the structure field with the duct wall. Within a channel, the flow is assumed to be one-dimensional in the axial direction, and the temperature field in the fuel, cladding, coolant, and structure is assumed to be two-dimensional in the radial and axial directions. Usually, a channel represents an average fuel element in a subassembly or a group of subassemblies. A channel may also represent pins in blanket or control subassemblies. Alternately, a single channel may also be used to represent the hottest pin in an assembly, or any other subset of a subassembly. The axial extent of a channel covers the entire length of a subassembly, including the core, the axial blankets, the fission gas plenum and the spaces above and below the pin/cladding geometry. Different channels may be used to account for radial and azimuthal design geometry, power, coolant flow, and burnup variations within the reactor core.

Table 1.3.1: Organization of the SAS4A/SASSYS-1 Manual

Chapter	Subject
1	Introduction (this chapter)
2	SAS4A/SASSYS-1 User's Guide
3	Pin Heat Transfer and Single-Phase Coolant Thermal/Hydraulics Model
4	Reactor Point Kinetics and Reactivity Feedback Models
5	PRIMAR-4: Primary and Intermediate Loop Thermal/Hydraulics Model
6	Plant Control and Protection Systems Model
7	Balance-of-Plant Thermal Hydraulics Model
8	DEFORM-4 Oxide Fuel and Cladding Mechanics Model
9	MFUEL Metallic Fuel Performance Model
10	SSCOMP Metal Fuel Characterization Model
11	FPIN2 Metal Fuel and Cladding Mechanics Model
12	TSBOIL Two-Phase Coolant Thermal/Hydraulics Model
13	CLAP Molten Cladding Dynamics Model
14	PLUTO2 Fuel-Coolant Interaction Model
15	PINACLE In-Pin Fuel Relocation Model
16	LEVITATE Fuel Relocation Model

Chapter 2 contains a general user's guide for SAS4A/SASSYS-1, including a complete description of the standard input file. Although Chapter 2 includes a summary description of every input parameter, it is essential that users consult the relevant chapters to understand the relationship between the input and the model formulations.

Chapter 3 contains the description of the formulation for the SAS4A/SASSYS-1 pin heat transfer and single-phase coolant thermal/hydraulics model. The subassembly-to-subassembly heat transfer model has been improved, and axial conduction in the coolant has been added. A sub-channel model has been introduced to provide accurate predictions of intra-assembly temperature and flow distributions.[1-16] This modeling addition is being validated with results from the EBR-II Shutdown Heat Removal Tests [1-17] as part of an International Atomic Energy Agency Coordinated Research Project.[1-18]

Chapter 4 contains the description of the formulation for the SAS4A/SASSYS-1 reactor point kinetics, decay heat, and reactivity feedback models. A new addition to this module is the ability to represent more detailed decay heat characteristics in multiple regions of the core. This module provides the reactor power level to the core thermal/hydraulics models for determination of the heating rate in the fuel, and receives core materials temperature and geometry information to calculate the reactivity feedbacks employed in the solution of the point kinetics equations.

Chapter 5 presents a full description of the formulation for the PRIMAR-4 sodium loops thermal/hydraulic model. This model provides boundary coolant pressure and flow conditions for the core channel models, including transient heat losses through normal and emergency heat removal systems and the transient performance of pumps. PRIMAR-4 includes the option for multiple core inlet and outlet coolant plena, permitting exact representation of the actual EBR-II coolant systems geometry. Compressible volumes in PRIMAR-4 may also be coupled with external computational fluid dynamics simulations to better represent flow and temperature distributions during transients.

The plant control and protection system model described in Chapter 6 is mostly unchanged from prior versions of SASSYS-1, except for the addition a sinusoidal function to represent oscillations in control-system signals.

The balance-of-plant (BOP) model described in Chapter 7 was implemented to permit 1) improved simulation of EBR-II design basis transients, 2) whole-plant analysis of IFR designs for optimization of advanced reactor control system strategies, and 3) core temperature margin assessments in unprotected accident sequences (i.e. beyond design basis accidents (BDBA) and anticipated transients without scram (ATWS)). In these latter sequences, core response depends strongly upon the performance of the balance-of-plant, because the core neu-

tronic and thermal/ hydraulic behavior is determined by the availability of heat sinks outside the core. The BOP model couples to PRIMAR-4 at the steam generator.

Chapter 8 provides a description of the DEFORM-4 fuel element behavior model for stainless steel-clad oxide fuel, which is unchanged from prior versions of SAS4A/SASSYS-1.

Chapter 9 contains the description of the MFUEL model, which treats the irradiation and transient behavior of U-Pu-Zr based metal fuel with HT-9 and D4 cladding. It includes physical phenomena unique to metallic fuel, such as fuel/cladding chemical interactions. The description of DEFORM-5, which was deprecated in Version 5.7, has been moved to the appendix of Chapter 9.

The SSCOMP model described in Chapter 10 reflects available metal fuel material properties evaluations recorded in the IFR Material Properties Handbook [1-19]. An efficient correlation technique has been implemented in all SAS4A/SASSYS-1 material properties routines that accurately generates the data from the IFR Handbook for use in all the modules of the code. It is planned to revise the material migration capability in SSCOMP for ternary fuel, to add models for fission gas generation and release, swelling, and all other phenomena needed to describe the transition from cold, clean, unirradiated conditions to hot irradiated conditions.

Chapter 11 contains the description of the FPIN2 metal fuel pin mechanics model [1-20]. FPIN2 is a validated model for metal fuel pin transient behavior. Unlike DEFORM-5, which treats only the cladding response, FPIN2 provides a finite-element solution of the fuel and cladding mechanics equations for the elastic/plastic response, including fission gas pressurization and migration, molten cavity formation and growth, and fuel/cladding chemical interaction and cladding thinning. The interface between SAS4A/SASSYS-1 and FPIN2 has been designed to permit stand-alone execution of FPIN2 for direct verification or to replace the FPIN2 thermal/hydraulics calculation with the SAS4A/SASSYS-1 counterparts for coupled calculations. The application for this model is design basis analysis of driver and experimental fuel elements in EBR-II for the purpose of margin-to-failure assessments.

The TSBOIL module for liquid metal coolant boiling and two-phase thermal/hydraulics calculations has been retained intact from previous versions of SAS4A/SASSYS-1. The current model includes a set of modifications to describe the sudden release of non-condensable fission gas from a cladding rupture in the upper fission gas plenum of metal fuel elements and the subsequent plenum blow-down and liquid coolant expulsion. This option has been used to assess the safety implications of long-term fuel element irradiations in EBR-II [1-21].

The CLAP and PLUTO2 models described in Chapter 13 and Chapter 14 are relevant only to oxide fuel, and have remained unchanged since the previous documentation.

The PINACLE model described in Chapter 15 and the LEVITATE model described in Chapter 16 have been upgraded for applications to metallic fuel [1-22]. The model enhancements added to PINACLE and LEVITATE for metal fuel include fuel/cladding and fuel/structure chemical interactions and fission gas generation and migration with fuel swelling. Preliminary analyses of TREAT M-Series in-pile metal fuel tests have been completed [1-23], and applications to severe accident sequences in metal-fueled IFR cores have been completed and documented [1-24].

1.4 References

1-1. Y. S. Tang, R. D. Coffield, Jr., R. A. Markley, “Thermal Analysis of Liquid Metal Fast Breeder Reactors,” Westinghouse Electric Corporation, Madison, Pennsylvania, American Nuclear Society Publications, 1978.

1-2. H. A. Bethe and J. H. Tait, “An Estimate of the Order of Magnitude of the Explosion When the Core of a Fast Reactor Collapses,” RHM-56-113, April 1956.

1-3. A. E. Waltar, A. Padilla, Jr., and R. J. Shields, “MELT-II, A Two-Dimensional Neutronics-Heat Transfer Program for Fast Reactor Safety Analysis,” Unpublished information, Hanford Engineering Development Laboratory, 1972.

1-4. K. K. Tabb, C. H. Lewis, L. D. O'Dell, A. Padilla, Jr., D. E. Smith, and N. P. Wilburn, "MELT-IIIB: An Updated Version of the MELT Code," Unpublished information, Hanford Engineering Development Laboratory, April 1979.

1-5. J. C. Carter, et al., "SAS1A, A Computer Code for the Analysis of Fast-Reactor Power and Flow Transients," ANL-7606, Argonne National Laboratory, October 1970.

1-6. F. E. Dunn, et al., "The SAS2A LMFBR Accident-Analysis Computer Code," ANL-8138, Argonne National Laboratory, October 1974.

1-7. M. G. Stevenson, W. R. Bohl, F. E. Dunn, T. J. Heames, G. Hoppner, and L. L. Smith, "Current Status and Experimental Basis of the SAS LMFBR Accident Analysis Code System," *Proc. Fast Reactor Safety Meeting*, American Nuclear Society, CONF-740401-P3, pp. 1303-3121, Beverly Hills, CA, April 2-4, 1974.

1-8. J. E. Cahalan, A. M. Tentner, and E. E. Morris, "Advanced LMR Safety Analysis Capabilities in the SASSYS-1 and SAS4A Computer Codes," *Proceedings of the International Topical Meeting on Advanced Reactors Safety*, American Nuclear Society, Vol. 2, pp. 1038-1045, Pittsburgh, PA, April 17-21, 1994.

1-9. F. E. Dunn, J. E. Cahalan, D. Hahn, H. Y. Jeong, "Whole Core Sub-Channel Analysis Verification with EBR-II SHRT-17 Test," *Proc. 2006 Int'l Congress on Advances in Nuclear Power Plants*, Reno, NV, June 6-8, 2006.

1-10. G. Birgersson, et al., Unpublished information, Argonne National Laboratory, 1988.

1-11. F. E. Dunn, et al., Unpublished information, Argonne National Laboratory, 1987.

1-12. C. E. Till, "Fast Reactor Directions: The LMR Integral Fast Reactor Program at Argonne," Proceedings of the International Fast Reactor Safety Meeting, Snowbird, Utah, August 12-16, 1990.

1-13. Argonne National Laboratory, Unpublished information, 1996.

1-14. J. Sackett, et al., "EBR-II Test Programs," Proceedings of the International Fast Reactor Safety Meeting, Snowbird, Utah, August 12-16, 1990.

1-15. T. H. Fanning, ed., "The SAS4A/SASSYS-1 Safety Analysis Code System," ANL/NE-12/4, Nuclear Engineering Division, Argonne National Laboratory, January 31, 2012.

1-16. F. E. Dunn, Unpublished information, Argonne National Laboratory, 1993.

1-17. D. Mohr, et al., "Loss-Of-Primary-Flow-Without-Scram Tests: Pretest Predictions and Preliminary Results," *Nuclear Engineering and Design*, **101**, 1987.

1-. T. Sofu and L. L. Briggs, "Benchmark Specifications for EBR-II Shutdown Heat Removal Tests," *Proc. Int'l Congress on Advances in Nuclear Power Plants*, ICAPP 2012, Chicago, IL, June 24-28, 2012.

1-19. IFR Properties Evaluation Working Group, "Handbook of Properties of Metallic Fuels," Argonne National Laboratory, Idaho Falls, Idaho, June, 1988.

1-20. T. H. Hughes and J. M. Kramer, "The FPIN2 Code - An Application of the Finite Element Method to the Analysis of the Transient Response of Oxide and Metal Fuel Elements," Conference on the Science and Technology of Fast Reactor Safety, British Nuclear Energy Society, Guernsey, May 12-16, 1986.

1-21. J. P. Herzog and K. J. Miles, Unpublished information, Argonne National Laboratory, 1989.

1-22. A. M. Tentner, Unpublished information, Argonne National Laboratory, April, 1993.

1-23. A. M. Tentner and Kalimullah, Unpublished information, Argonne National Laboratory, 1993.

1-24. E. E. Morris and A. M Tentner, Unpublished information, Argonne National Laboratory, 1994.

CHAPTER

TWO

SAS4A/SASSYS-1 USER'S GUIDE

2.1 Introduction

This chapter contains information that will aid the user in understanding the program architecture of the SAS4A/SASSYS-1 computer code, and the relationship between its modeling concepts and the program logic and data flow. The code structure is described in terms of modules and subroutines, as are the data management features employed to communicate input and calculated data among modules. The contents and organization of input and output files are described, and results from simplified example problems are included and illustrated.

2.2 Modeling Concepts and Code Structure

2.2.1 Code Structure Basis

The code structure of SAS4A/SASSYS-1 is constructed to reflect the physical modeling assumptions employed. In the core models, the basic geometric modeling element is a fuel pin, its cladding, and the associated coolant and structure, with the structure field representing wire wraps, grid plates, and/or hex cans. In the SAS4A/SASSYS-1 terminology, the term "channel" is used to denote collectively this basic element of fuel, cladding, coolant, and structure. In a single-pin model, a single average channel is used to represent the average of many pins in the reactor, and multiple channels are used to extend the model to all the pins in the reactor. In a multiple-pin model, each channel represents one or more pins in a subassembly, and multiple-pin subassembly models are joined with single-pin subassembly models to cover the whole reactor core. A single SAS4A/SASSYS-1 channel may therefore represent either one pin, or a large number of pins in many subassemblies. In either case, the elementary unit from a code structure and data management stand-point is an individual channel.

The code structure of SAS4A/SASSYS-1 is also the result of the programming language employed and the functional requirements of the phenomenological models. The programming language used for SAS4A/SASSYS-1 is ANSI FORTRAN, and the organization of the code follows the FORTRAN convention of a MAIN program with a number of subroutines and functions. For the purpose of this discussion, the subroutines and functions of SAS4A/SASSYS-1 are grouped according to purpose into one of the sixteen modules listed in Table 2.2.1. These modules are aligned in a one-to-one fashion with the phenomenological models of SAS4A/SASSYS-1, each of which is described by a chapter in this report. (The D3IF module is not documented in this report.)

Table 2.2.1: SAS4A/SASSYS-1 Modules

Module	Purpose
ROOT	Logic path control, data management, and material properties services.
TSCL0	Single phase liquid coolant thermal/hydraulics and fuel element heat transfer.
TSPK	Reactor point kinetics and first order perturbation theory reactivity feedbacks.
PRIMAR-4	Primary and secondary coolant loops and components thermal/hydraulics and heat transfer.
CNTLSYS	Reactor and plant control and protection systems simulation.
BOP	Balance-of-plant systems and components thermal/hydraulics and heat transfer.
DEFORM-4	Oxide fuel/cladding fuel element mechanics.
MFUEL	Metallic fuel/cladding fuel performance model.
SSCOMP	Metallic-fuel material properties.
FPIN2	Metallic fuel/cladding fuel element mechanics.
TSBOIL	Two-phase (boiling) coolant thermal/hydraulics and fuel element heat transfer.
CLAP	Molten cladding relocation and heat transfer.
PLUTO2	Post-cladding-failure oxide fuel/liquid coolant interactions with fuel/coolant thermal/hydraulics.
PINACLE	Molten metallic fuel relocation and heat transfer prior to cladding failure.
LEVITATE	Post-cladding-failure oxide and metallic fuel relocation with fuel/cladding heat transfer.
D3IF	Interface to DIF3D-K for TSPK input data generation or DIF3D-K space/time neutronics.

The SAS4A/SASSYS-1 MAIN program calls three subroutines in a module that perform 1) data management initialization, 2) code execution, and 3) data management termination. A flowchart for the MAIN program is shown in Figure 2.2.1. Figure 2.2.2 is a flow diagram for INPDRV, the input driver routine. Figure 2.2.3 and Figure 2.2.6 present flow diagrams for SSTHRM and TSTHRM, the steady-state and transient driver subroutines. The contents of the channel loop in TSTHRM are shown in Figure 2.2.9.

2.2.2 Standard Input Data Reading

INPDRV calls subroutine READIN, which reads card images from the ASCII input file (see Section 2.5). A complete description of the contents of the input file is contained in Section 2.7. If called for, READIN will call subroutine RESTAR to read a binary restart file (RESTART.bin). Subroutine READIN prints input data card images as they are read. Once all input has been read, subroutine DATOUT prints a copy of the final input deck as card images. If control system input has been provided, subroutine CTLIN3 will perform checking of the input data. Subroutine PMCHEK performs a similar function for input material property data. If specified, subroutine INPEDT will print a formatted version of the input data blocks. Subroutine SSIN01 initializes a number of constants used in the calculations and subroutine PRECAL prepares correlations of metallic fuel material properties (density, specific heat, and thermal conductivity) based on the Integral Fast Reactor Handbook [2-1, 2-2], if specified.

A developmental interface to the DIF3D neutron diffusion theory code [2-3] is provided for by a call to subroutine DATEDT. When fully operational, DATEDT will manage the calculation of the reactor power distribution, reactivity feedback, and point kinetics data for the TSPK module. (Future plans call for implementation of the DIF3D-K space-time neutronics code [2-4] in SAS4A/SASSYS-1). Finally, INPDRV calls CRVSUP to supervise the fitting of various input data tables to correlations.

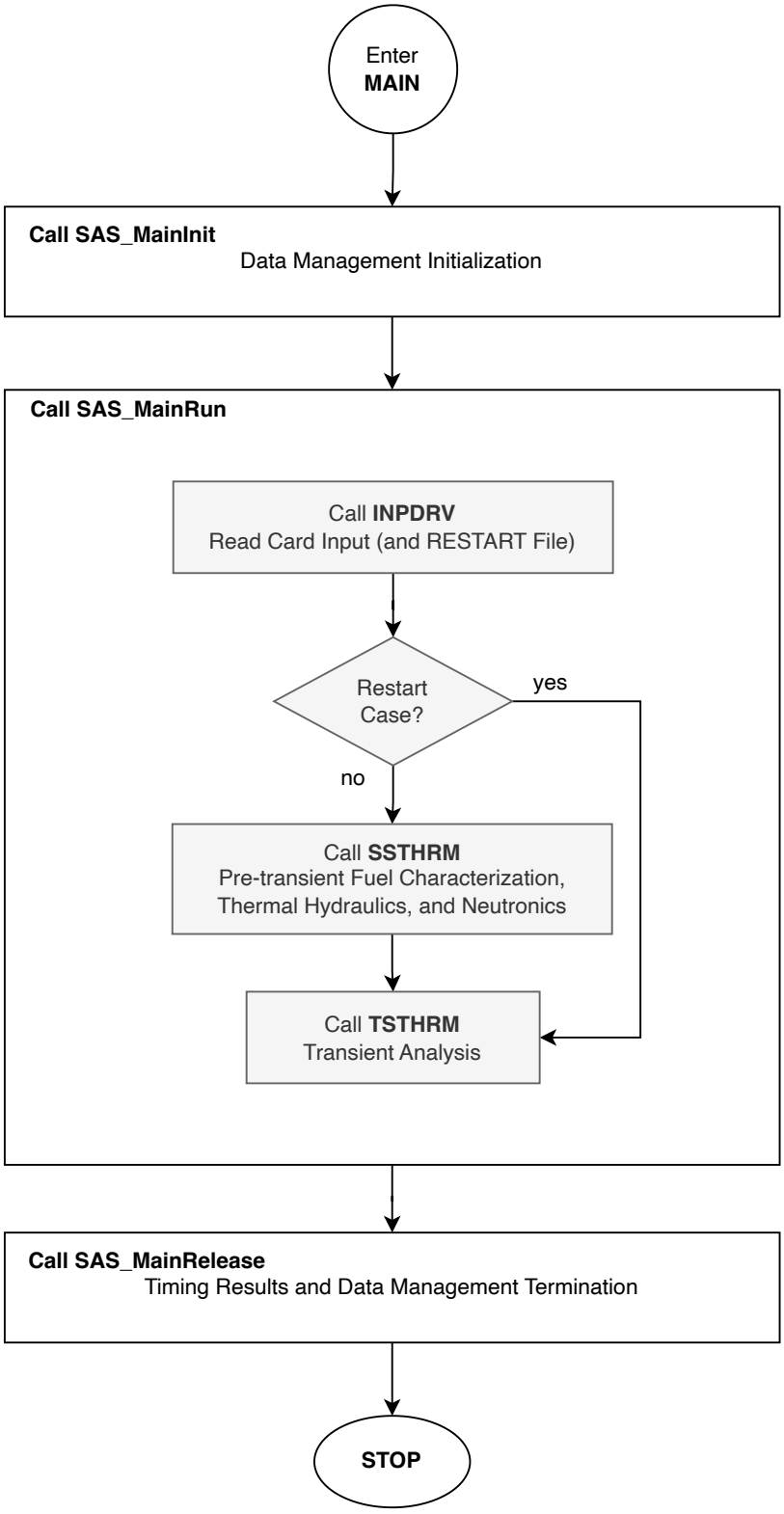


Figure 2.2.1: MAIN Program Flow Diagram

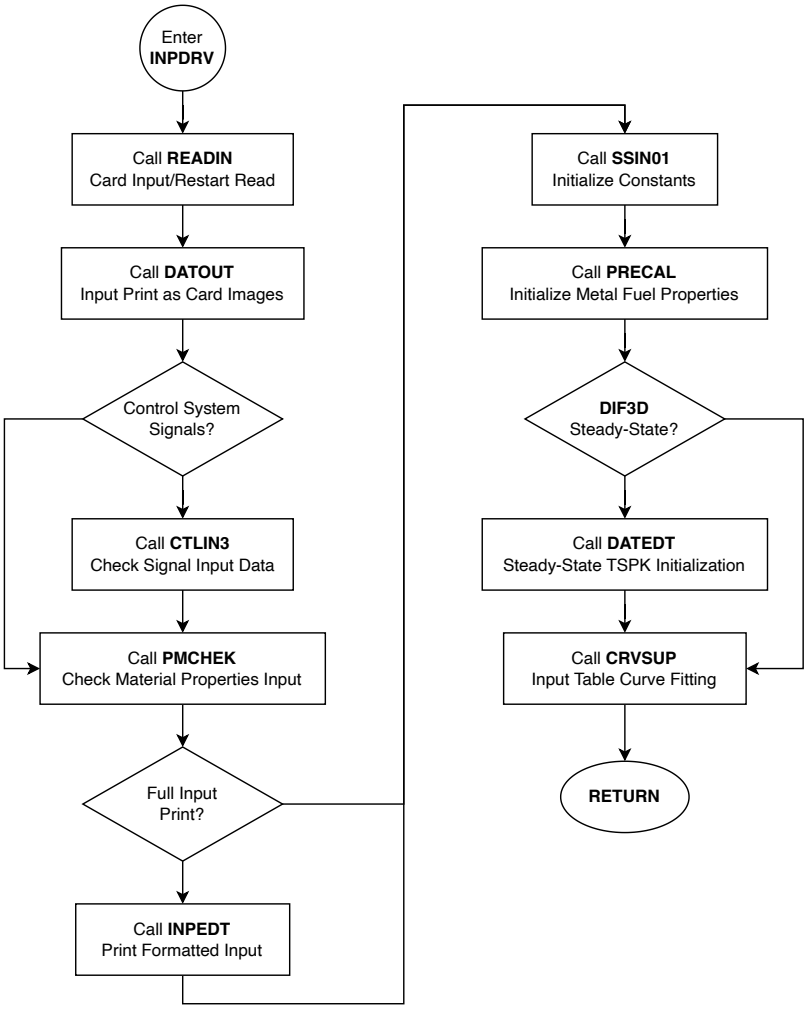


Figure 2.2.2: INPDRV Subroutine Flow Diagram

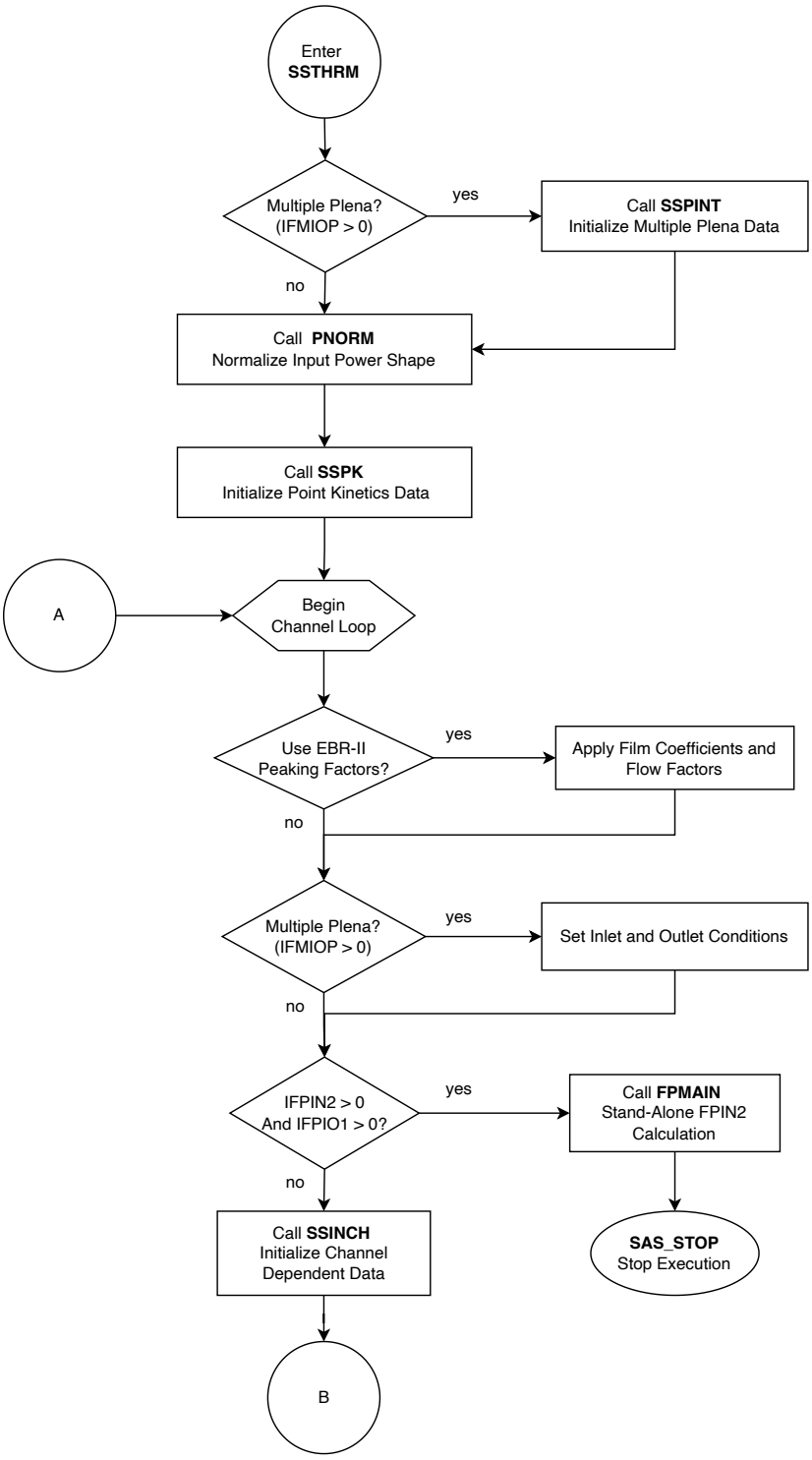


Figure 2.2.3: SSTHRM Subroutine Flow Diagram Part 1

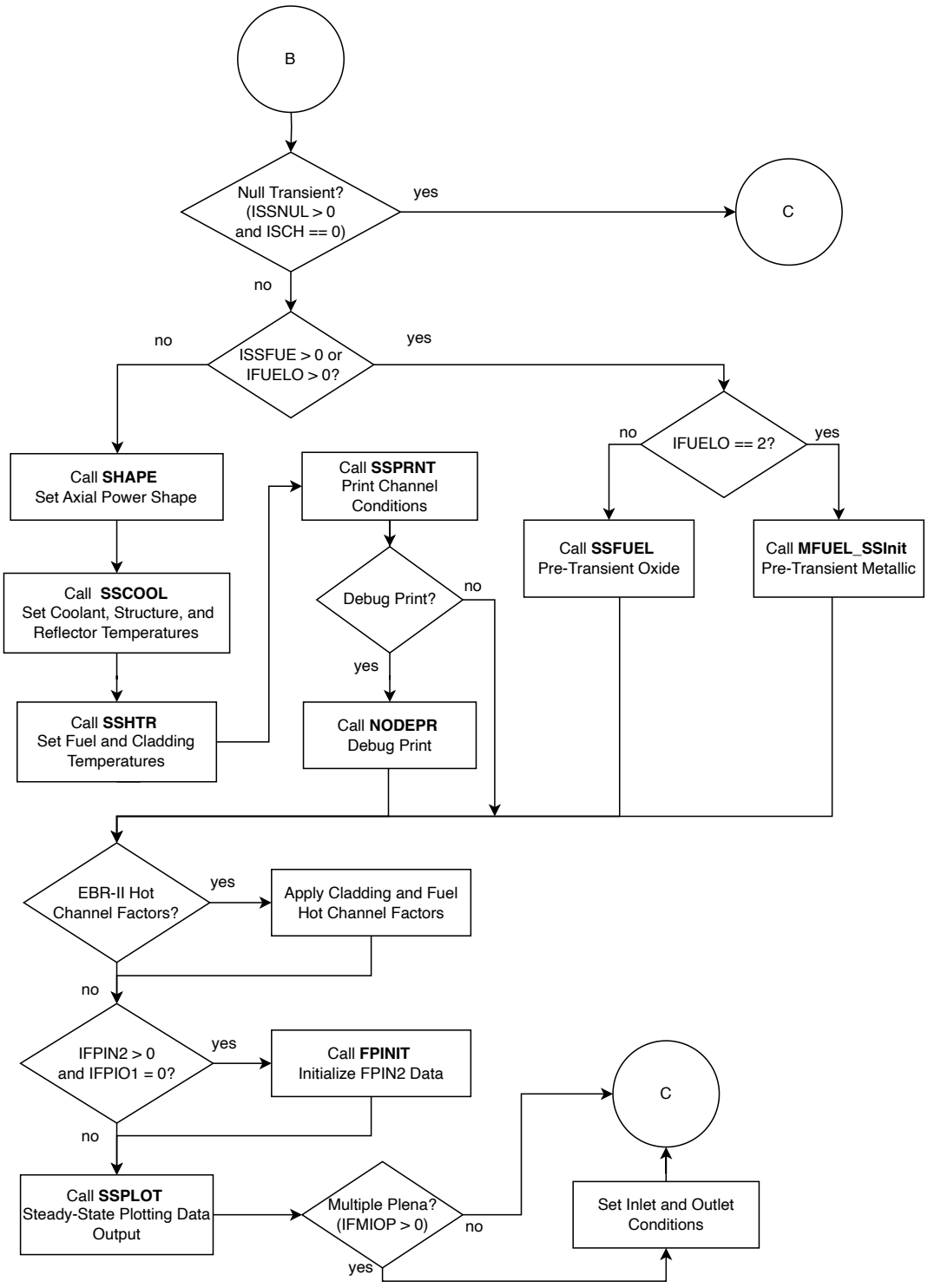


Figure 2.2.4: SSTHRM Subroutine Flow Diagram Part 2

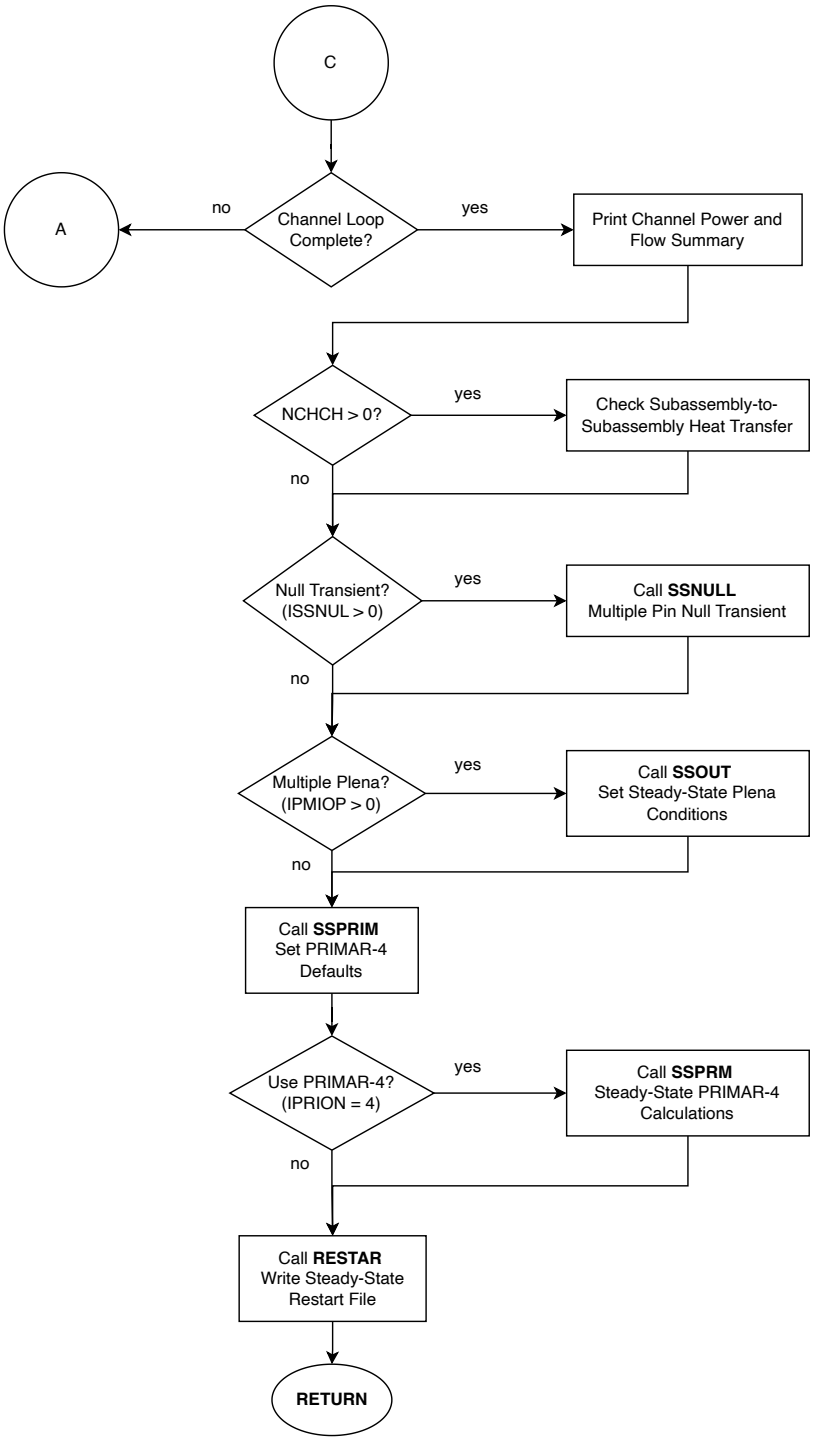


Figure 2.2.5: SSTHRM Subroutine Flow Diagram Part 3

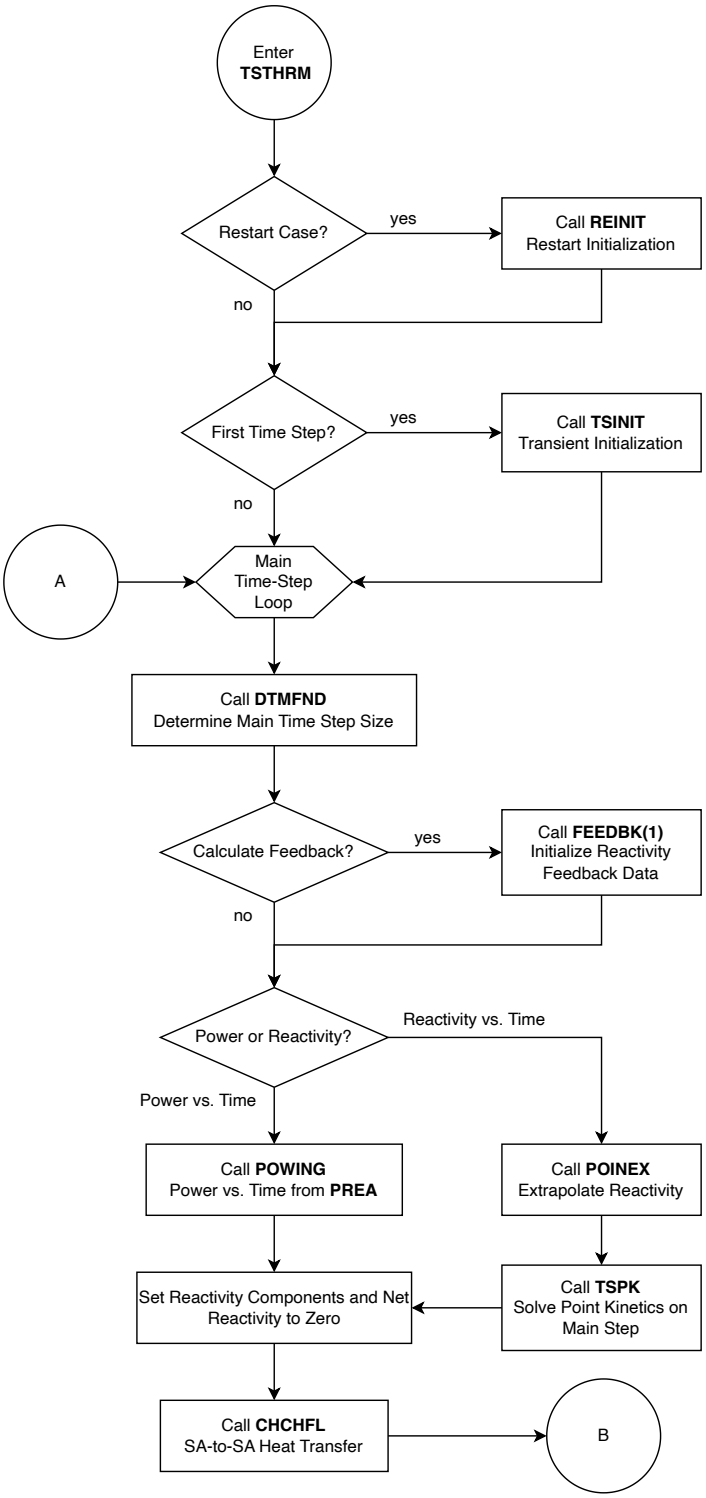


Figure 2.2.6: TSTHRM Subroutine Flow Diagram Part 1

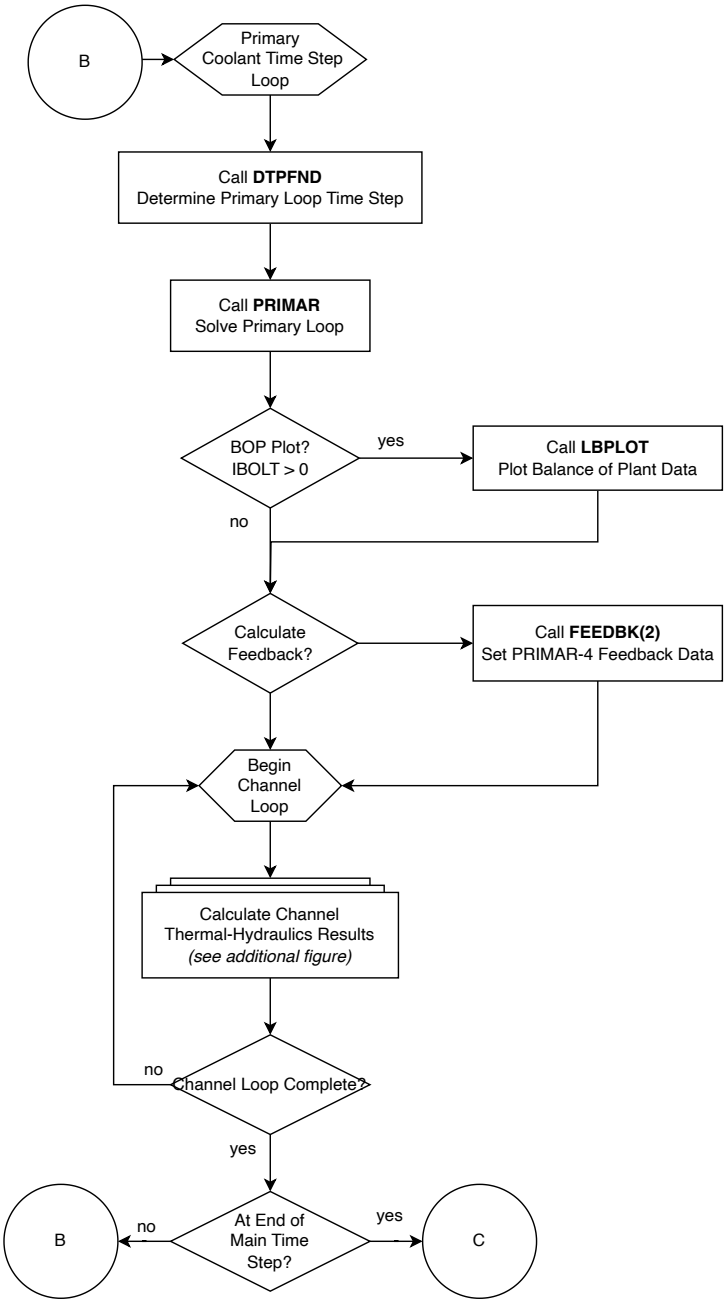


Figure 2.2.7: TSTHRM Subroutine Flow Diagram Part 2

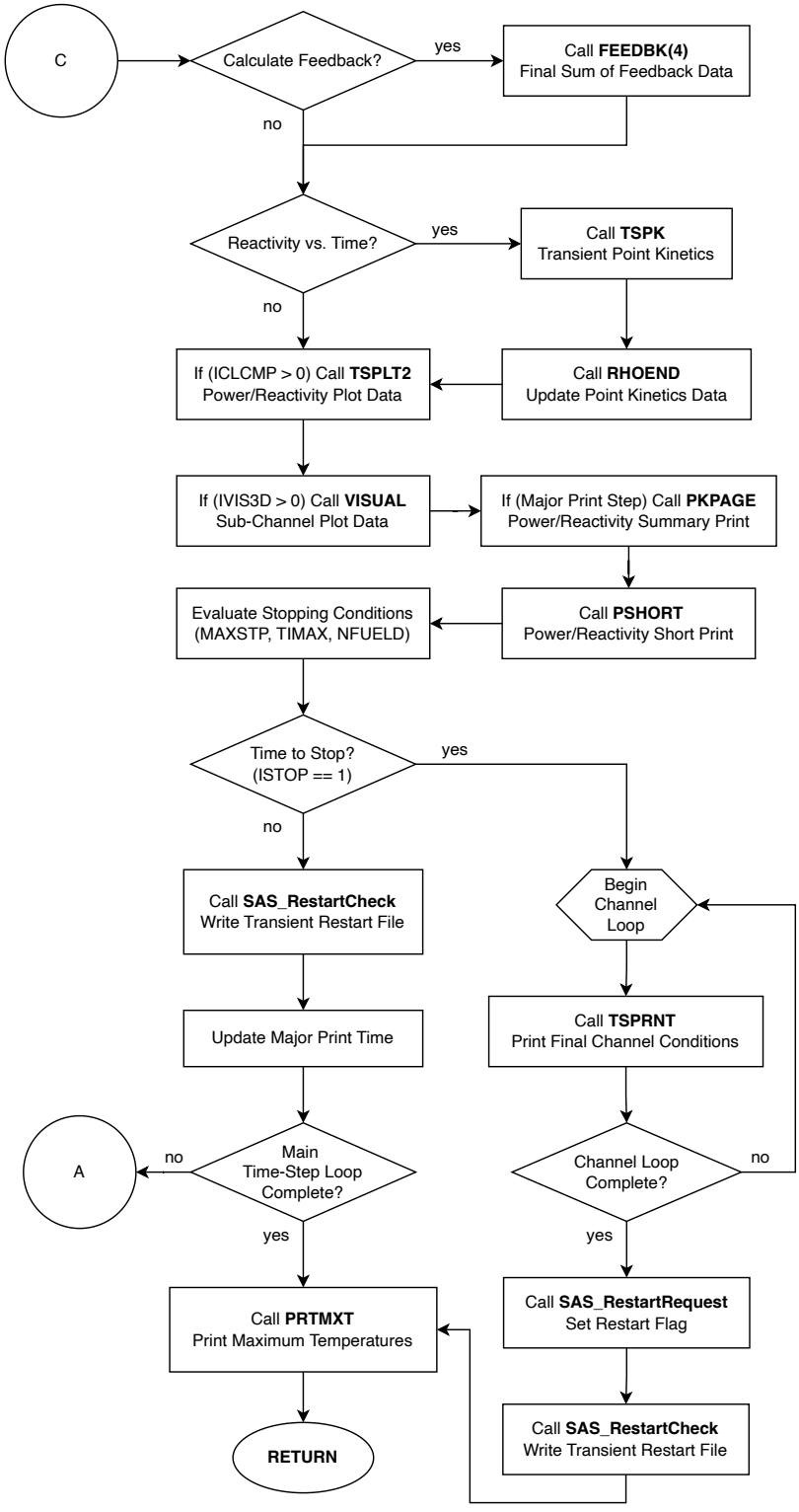


Figure 2.2.8: TSTHRM Subroutine Flow Diagram Part 3

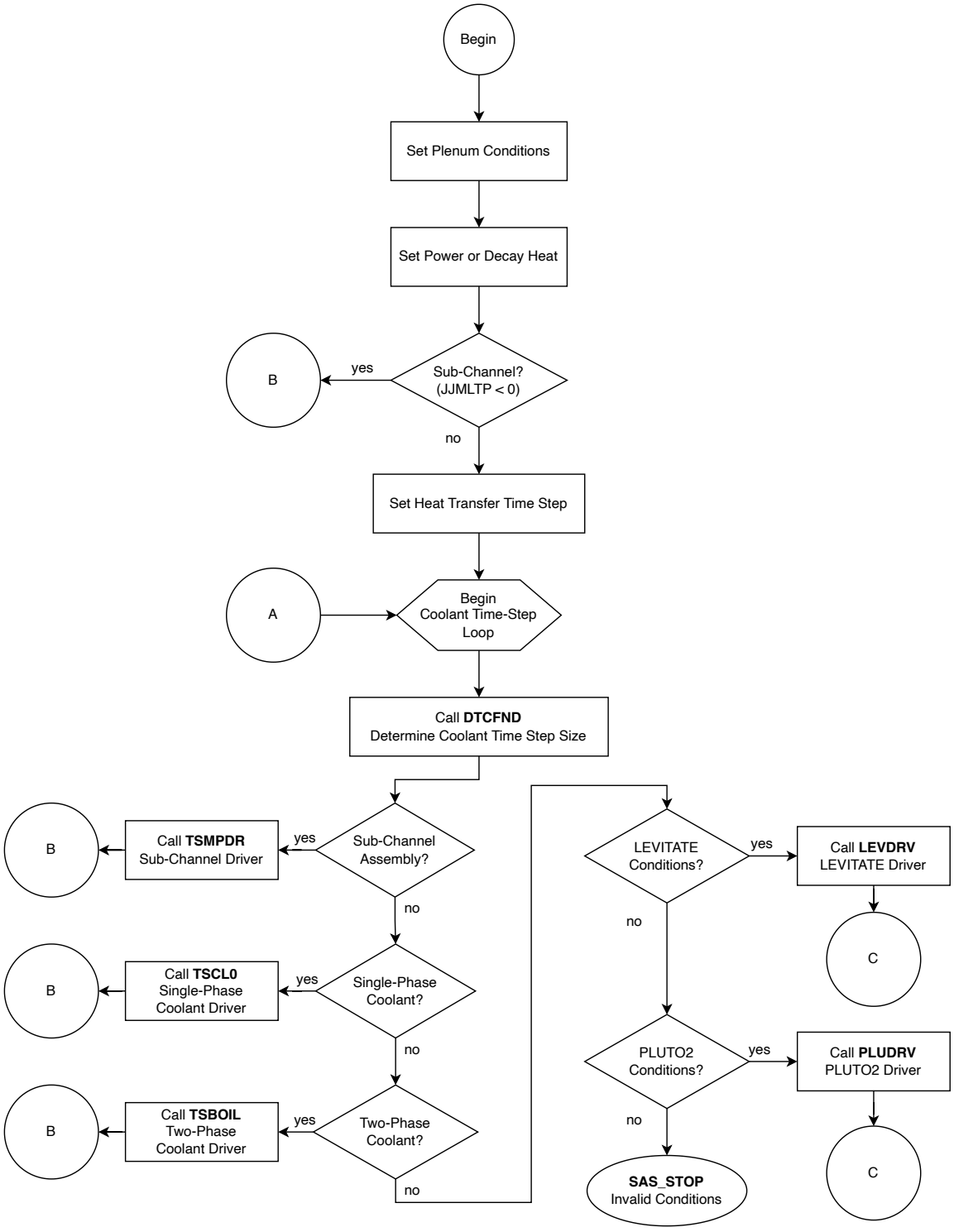


Figure 2.2.9: Flow Diagram for the Channel Loop in TSTHRM Part 1

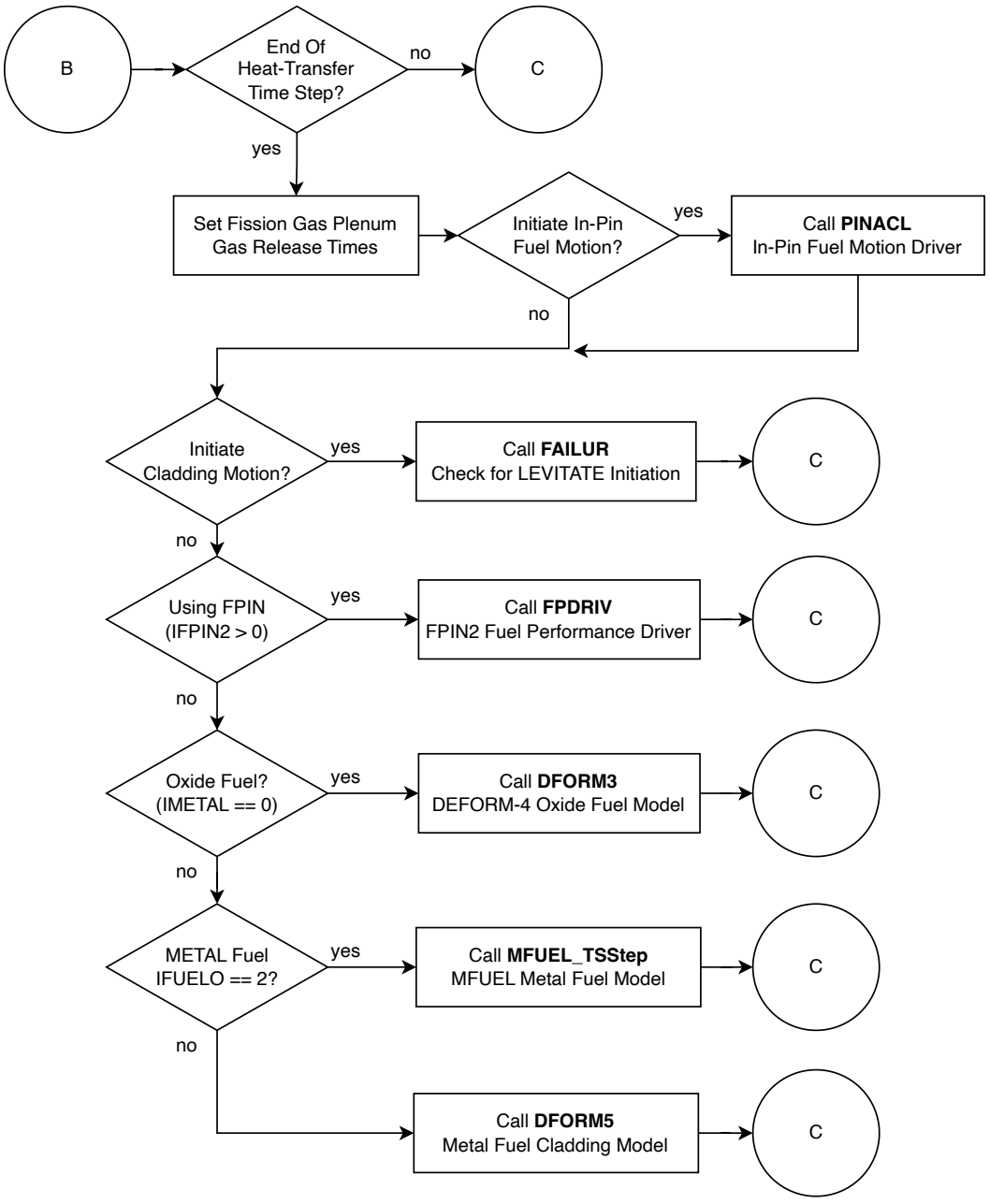


Figure 2.2.10: Flow Diagram for the Channel Loop in TSTHRM Part 2

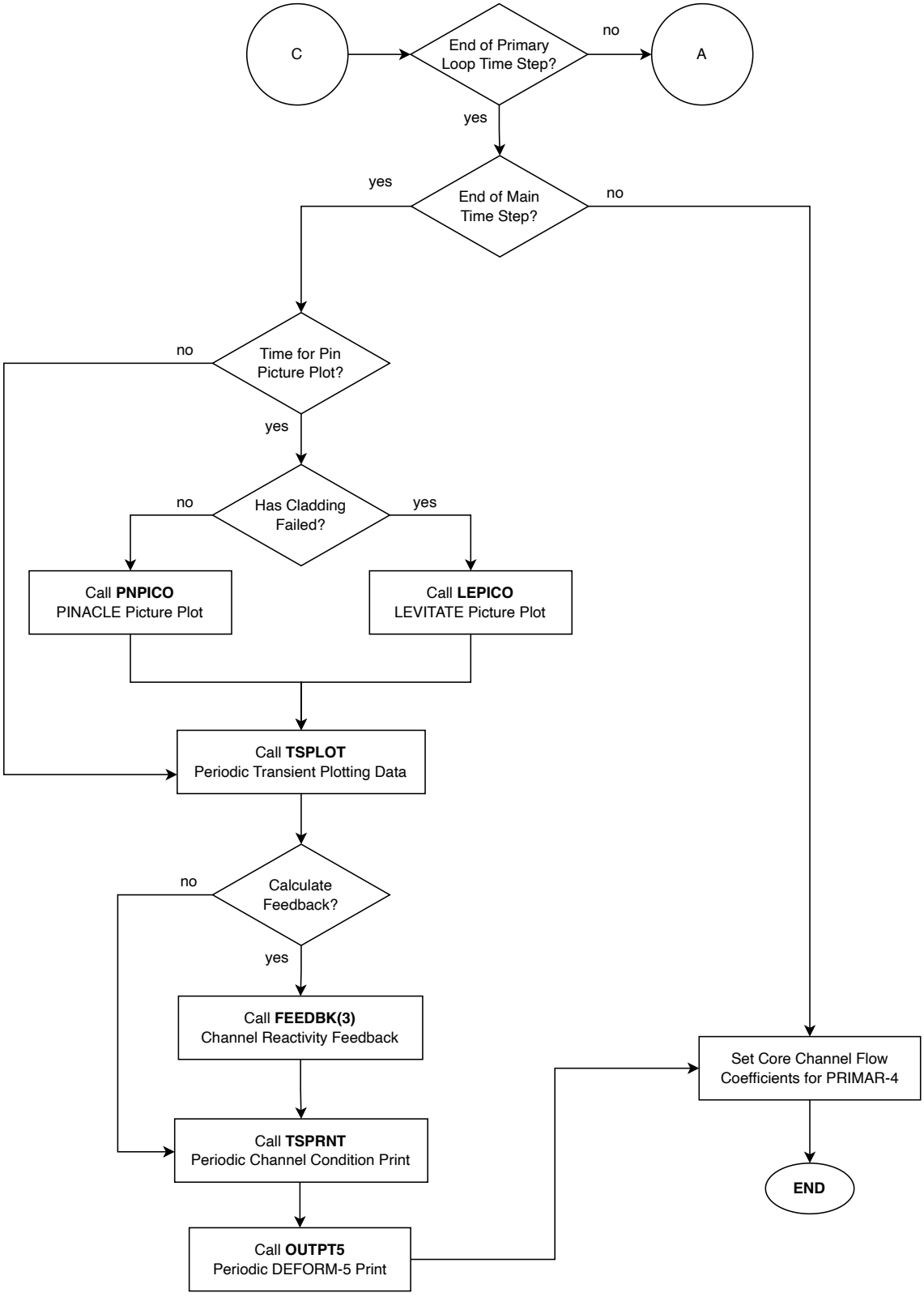


Figure 2.2.11: Flow Diagram for the Channel Loop in TSTHRM Part 3

2.2.3 Steady-State Calculation

Figure 2.2.3 presents a flow diagram for the steady-state (pre-transient) driver subroutine, SSTHRM. SSTHRM begins by making calls to SSPINT, which initializes data for the multiple-plenum option in PRIMAR-4, to PNORM, which normalizes the reactor power distribution to the specified condition, and to SSPK, which initializes data for the point kinetics solution module, TSPK. Within a loop over all channels, SSTHRM performs optional initializations for applications of EBR-II thermal/hydraulic uncertainty factors and the multiple-plenum option. Next, an optional path to a single-channel, stand-alone execution of the FPIN2 computer code [2-5] is provided. Subroutine SSINCH performs channel-dependent data initialization, and subroutines SSFUEL, and MFUEL_SSInit manage the pre-transient thermal, hydraulic, and mechanics calculation for oxide (DEFORM-4) and metallic (MFUEL) fuels, respectively. Should the fuel performance option not be elected, SSTHRM calls a) subroutine SHAPE to set the axial channel power distribution, b) subroutine SSCOOL to calculate the coolant, structure, and reflector temperatures, c) subroutine SSHTR to calculate the fuel and cladding temperatures, d) subroutine SSPRNT to print the steady-state temperature, geometry, and neutronics results, and e) subroutine NODEPR for an optional diagnostic print of the fuel/cladding temperature calculation. Once all temperatures have been determined, an option is available to apply hot channel factors. If the FPIN2 model is specified for the current channel, subroutine FPINIT provides the interface to the SAS4A/SASSYS-1 temperature calculation. Subroutine SSPLOT provides an entry for the writing of channel-dependent, steady-state results for subsequent plotting. The final action in the channel loop is optional initialization of data for the multiple-plenum model. Following the channel loop, a summary of the reactor power and flow data for all channels is printed, and the optional subassembly-to-subassembly input data is checked for consistency. If the multiple-pin subassembly model has been invoked, subroutine SSNULL performs a constant power and flow transient calculation to bring the participating channels to an equilibrium temperature condition. If the multiple-plena PRIMAR-4 option is used, steady-state plena conditions are then set by a call to subroutine SSTOUT. PRIMAR-4 default values are set in a call to SSPRIM, and the optional primary loop, secondary loop, and balance-of-plant steady-state conditions are set by a call to subroutine SSPRM4. Finally, SSTHRM calls subroutine RESTAR to write a restart file RESTART.dat defining the steady-state condition.

2.2.4 Transient Calculation

The flow diagram for subroutine TSTHRM, the transient calculation driver routine, is given in Figure 2.2.6. If the problem is beginning from a restart file, subroutine REINIT is called to perform necessary restart initialization. Next, subroutine INGRFN is called to initialize transient driving conditions from the shared memory segment for the simulator option.

The transient problem time domain is divided into time steps as depicted in Figure 2.2.12. Reactivity feedbacks and solutions to the point kinetics equations are obtained on the main time steps. The primary loop, secondary loop, and balance-of-plant thermal/hydraulics solutions are obtained on the primary loop time step, which is a sub-step of the main time step. Recalculations of the core channel temperatures are carried out on the heat transfer time steps, and solutions of the channel hydraulics equations are obtained on the coolant time steps. The heat transfer time step is a sub-step of the main time step, and the coolant time step is a sub-step of the primary loop time step. A coolant time step may not span the end of a heat transfer time step. Heat transfer and coolant time steps may vary from channel to channel for single-pin modeling, or from subassembly to subassembly for multiple-pin or sub-channel modeling. The time step accounting algorithm maintains consistency among the time step levels while adjusting the various step lengths to preserve power, reactivity, and temperature change limits as set by default or in the input to preserve accuracy.

Before the first transient main time step, subroutine TSINIT is called to perform initialization. Each main time step begins with a call to subroutine DTMFND to set the time step length, and subroutine FEEDBK is called to initialize the reactivity feedback calculation. For the power vs. time option, subroutine POWING is called to set the reactor power variation over the time step; for the reactivity vs. time option, subroutine POINEX extrapolates the reactivity and subroutine TSPK calculates the reactor power. If the subassembly-to-subassembly heat transfer

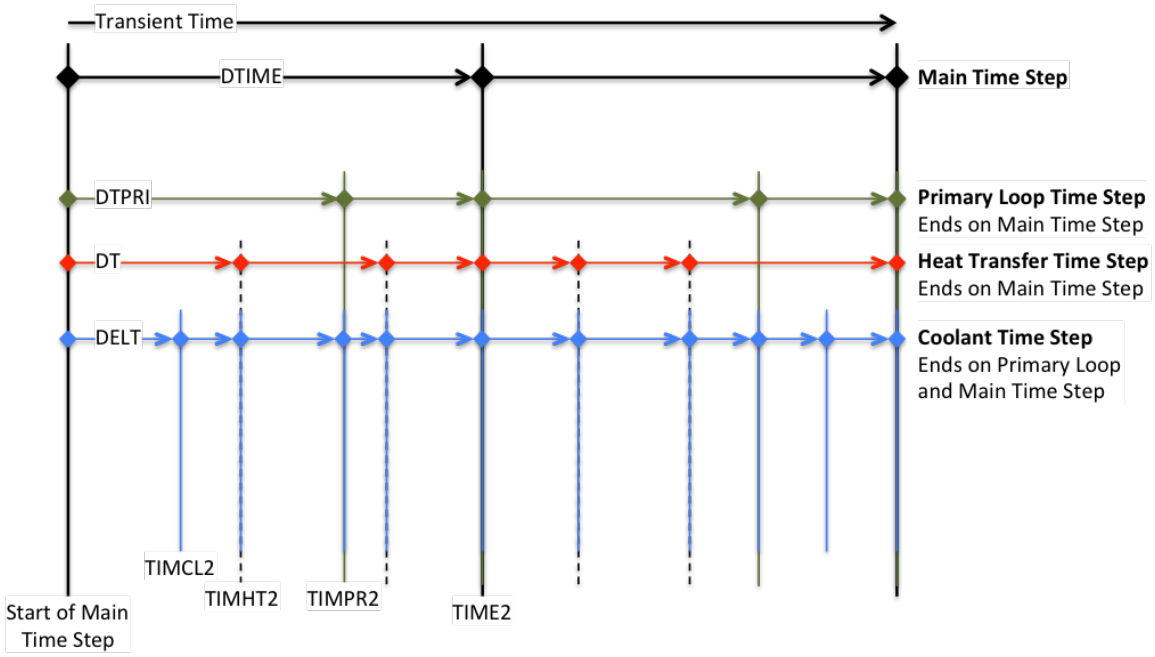


Figure 2.2.12: SAS4A/SASSYS-1 Time Step Hierarchy

option has been specified, subroutine CHCHFL is called to calculate the channel-to-channel heat fluxes. The primary loop time step, a substep of the main time step, begins with a call to subroutine DTPFND, which sets the primary loop time step length. Subroutine PRIMAR supervises the thermal/hydraulic calculations for the primary and secondary loops and the balance-of-plant, setting temperature and flow boundary conditions for the core. Subroutine LBPLOT provides optional writing of selected balance-of-plant plotting data on the last primary loop time step in a main time step, and subroutine FEEDBK then retrieves PRIMAR-4-calculated data for calculation of reactivity feedbacks. At this point, a loop over channels is begun. Optional multiple-plena conditions are set, as is the channel decay heat or power curve. At this point, if the current channel is one of the pins in the multiple-pin subassembly model, but not the first channel, then a skip is made around the subsequent channel thermal/hydraulics driver subroutines (i.e. TSCL0, TSBOIL, LEVDRV, and PLUDRV), since this channel's thermal hydraulics solution has already been obtained within the multiple-pin model. (At present, the multiple-pin model is limited to single-phase coolant flow, but may be extended to coolant boiling and pin disruption modeling in the future). On the other hand, if this is a single-pin channel or the lead multiple-pin channel, subroutine DTCFND is called to set the coolant time step length (a subset of the primary loop time step and variable from one channel to another in the single-pin model and from one subassembly to another in the multiple-pin model). The channel coolant hydraulics solution is obtained on the coolant time step. TSTHRM then calls one of the channel thermal/hydraulics driver subroutines (TSCL0, TSBOIL, LEVDRV, or PLUDRV), depending on the thermal/hydraulics conditions in the channel. One of these driver subroutines then advances the channel calculation to the end of the current coolant time step, checking the current time along the way to determine whether a recalculation of the pin temperatures is necessary.

At this point in subroutine TSTHRM, a check is made to determine if the current coolant time step has ended on a heat transfer time step. If not, the calculation proceeds to the next coolant time step. If the end of the heat transfer has been reached, newly-determined channel temperatures are available, and this triggers tests on the need for fuel element mechanics (DEFORM-4, MFUEL, DEFORM-5, FPIN2) or in-pin fuel relocation (PINACLE) calculations. But first, a test is made to determine whether the input has specified the approach of a fission-gas-plenum cladding failure time, and subsequent gas release into the coolant channel. Next, a check is made to determine whether this channel has experienced cladding failure; if so, subsequent tests for fuel pin mechanics or in-pin fuel relocation are skipped. If cladding failure has not occurred, the PINACLE module is

invoked if conditions are met for in-pin fuel relocation, and subroutine FAILUR is called to check for cladding failure. If one of the fuel pin mechanics options has been specified, it is executed at this point. This brings the calculation to the bottom of the primary loop and main time step branches.

At the end of a primary loop time step, a check is made to determine whether the end of the main time step has been reached. If not, another primary loop time step will be made to advance the calculation. However, if the end of the main time step has been reached, a series of tests are made to check for the writing of plotting file data (subroutines PNPICO, LEPICO, and TSPLOT), to sum a channel's contribution to the net reactivity (subroutine FEEDBK), and to print channel conditions (subroutines TSPRNT and OUTPT5). A test is made to determine whether all channels have been advanced to the end of the primary loop time step; if not, the next channel is calculated, but if so, a test is made on the current time to check for the end of the main time step. If the end of the main time step has not been reached, subsequent primary loop time steps are made until the end of the main time step is found. Then, subroutine FEEDBK is called to total the net reactivity, and for the reactivity vs. time option, subroutines TSPK and RHOEND are called to solve the point kinetics equations over the main time step. If specified, the reactor power and reactivity data are added to the plotting file (entry TSPLT2), and subroutine PKPAGE is called to print the power and reactivity summary page on a time step interval set by input. Subroutine PSHORT produces a short-form power and reactivity print on each time step, followed by tests for problem termination on the basis of computing time limit, maximum number of time steps, maximum problem time, or minimum fuel motion reactivity. If a termination condition is found, a test is made to produce the major channel prints (subroutine TSPRNT) in the event they were not made on the current time step. If a termination condition is not found, the calculation continues following checks for a) the writing of restart file from subroutine RESTAR, b) the optional transfer of data to and from the shared memory segment in conjunction with the simulator option, and c) the optional print of a computing time summary. At termination, the restart file is written automatically and the final computing time summary is printed, followed by the return to program MAIN, where a summary of all subroutines entered in the calculation is printed and execution halts.

2.3 Data Management

In order to minimize memory requirements and enhance computational performance, the SAS4A/SASSYS-1 data management scheme maintains a framework for optional data storage. Accordingly, users have the ability to control data allocation for specific modules via the storage allocation record. These storage flags affect the creation of module-specific data structures that are utilized for management of dynamically allocated memory and common block data.

Flag IADEFC on the storage allocation record affects channel-specific data allocation for DEFORM-4, DEFORM-5, and FPIN2 data, meaning allocation of these data structures is necessary if any of these modules are to be utilized. Flag IAPLUC controls allocation of memory for data related to PINACLE, LEVITATE, or PLUTO2; similar to IADEFC, allocation using IAPLUC is necessary if any the related modules are utilized during a simulation. It should be noted that problems that initiate either PINACLE, LEVITATE, or PLUTO2 must have either DEFORM-4 or DEFORM-5 also running, but the DEFORM modules may operate without PINACLE, LEVITATE, or PLUTO2.

Two additional storage areas are allocated for the control system module (CNTLSYS) and the balance-of-plant module (BOP). These areas need not be allocated if the corresponding module is not to be executed. Allocation of the CNTLSYS storage is controlled with variable IACNTL on the storage allocation record, and the BOP module storage area allocation option is controlled with variable IALBOP in the same record.

If the optional SAS4A/SASSYS-1 interface to the DIF3D code is used, the DIF3D modules allocates two storage areas under the control of the BPOINTR package. These storage areas are used only on the DIF3D side of the interface.

2.4 Supported Platforms

Currently, SAS4A/SASSYS-1 is supported on 64-bit macOS, Linux, and Windows platforms with the x86_64 architecture.

2.5 Input and Output Files

2.5.1 SAS Input File

In the past, SAS4A/SASSYS-1 input was limited to 80-column, fixed-formatted card images. This is still the basic format used by most SAS input files. However, Version 5.3 introduces the option to use free-formatted input of any length for integer and floating-point input blocks. A full description of the contents of the input file is contained in Section 2.7.

In all SAS input files, the first three (non-comment) records of the input file are required. The first two records each contain 72 columns of problem title data that will appear at the top of each page of the output file. Columns 73-80 of the first card image must contain, left adjusted, the major and minor version number of the SAS code being executed. The third record is the storage allocation card, which provides SAS information needed to allocate the various data containers used for input processing. If the DIF3D interface option is invoked, an additional storage allocation card (fourth record) is required.

If the current execution is to be restarted from a prior execution, the input file must contain a RESTART directive to invoke the reading of a restart file from RESTART.bin (see Section 2.5.3). The reading of the restart file takes place at the point the restart record is encountered on the input file. Input blocks loaded before the restart record will be over-written by the restart file data, and input blocks following the restart record will modify or replace data read from the restart file.

The input data blocks for SAS are described in Section 2.7. Each input data block begins with a block identifier record and ends with a block delimiter record. The block identifier record contains the block name and number, and two integers denoting the relevant current channel and the type of background data fill to be supplied before data reading begins. (The background may be “zeros” or the same block entered previously for this or another channel.) The block delimiter record accepts either a negative integer one (“-1” right adjusted) entered in the first I6 field of the record, or a left adjusted “END” statement.

Within the body of a SAS input file, INCLUDE directives may be used to read additional input data from external files. Any number of nested INCLUDE directives may be used, but circular references are not allowed. Support for INCLUDE directives was added in Version 5.3.

Reading of input is terminated by an ENDJOB directive. Any records following the ENDJOB record are ignored.

2.5.2 SAS Output Files

SAS4A/SASSYS-1 prints ascii output as 133-column, formatted, printed line images. Listings of example output files can be found in the example problems output listings provided with the code distribution.

The ascii output file is occasionally divided by logical page breaks. (A fixed page size of 60 lines is no longer maintained.) Each logical page break is headed with the two title records that begin the input deck. The title lines also contain code version identification, the page number, job and user identification, the execution date, and the clock time. Pages containing channel-dependent information will have the channel number printed in the upper left-hand corner.

The ascii output file begins with a print of information from the header, including primary and secondary titles and memory allocation flags. During input processing, each record from the input file is recorded to the file SAS.log. An optional compiled version of the input may be written to INPUT.dat (see IPDECK). An optional, formatted and annotated print of the input data is available, followed by a tabular print of coolant material properties. This is followed by printer plots of curves formed by the tabular fitting procedure.

The steady-state results are printed next, beginning with prints of channel-dependent data. The prints vary depending on the computational modules invoked, but usually contain node-wise masses, porosities, and temperatures, as well as radial and axial pin geometry. Also printed are nodal powers and reactivity feedback coefficients. Following the channel-dependent prints, steady-state results from the primary loop (PRIMAR-4), control system, and balance-of-plant may appear, if those modules are employed.

A short print of time, power, and reactivity is produced on each main transient time step. The exact form of this print varies depending on whether the default reactivity feedback routines are employed or the EBR-II feedback routines are used. On specified main time steps, long prints of the current channel conditions (geometry, pressure, temperature, etc.) are produced. In addition, each module may produce intermittent transient prints of calculated results. These prints are controlled by input data. Extensive diagnostic prints may also be triggered through input for most of the computational modules.

SAS4A/SASSYS-1 also prints binary output files. These files and their contents are summarized in the following sections.

2.5.2.1 CHANNEL.dat

CHANNEL.dat is output if requested by the user (**ICLCMP**). At a user-defined frequency, see **MSTPL1**, channel-dependent and core wide data is written to the CHANNEL.dat file. The CHANNEL.dat file is an unstructured binary file. Converter utilities are provided as part of the distribution package. The converter utilities translate the binary data into a CSV or XML format.

Each time data is written to the binary file a row is created for each channel that contains the data presented in Table 2.5.1. Following the channel-dependent entries is an additional row containing the core-wide data, presented in Table 2.5.2, is written.

Data that has a corresponding CSV column number is converted by the CHANNELtoCSV tool distributed with the code.

Table 2.5.1: Channel-Dependent Contents of CHANNEL.dat

Binary Index	CSV Column	Data Type	Description
1		Integer	Time step number
2	Appears in CSV file name	Integer	Channel number
3		Integer	KMAX
4	2	Double	ZFuel - Fuel elevation at **JCLN** or Peak (m)
5	3	Double	ZClad - Clad elevation at **JNEN** or Peak (m)
6	4	Double	ZCool - Coolant elevation at **JNCN** or Peak (m)
7	5	Double	ZStruct - Structure elevation at **JNSN**, if specified (m)
8	6	Double	Fuel elevation at **JRPRO**, if specified (m)
9	7	Double	Average coolant temperature at ZCool (K or C)
10	8	Double	Clad temperature at ZClad (K or C)
11	9	Double	Fuel temperature at ZFuel (K or C)
12	10	Double	Coolant saturation temperature at ZCool (K or C)
13	11	Double	Fuel melt fraction (-)
14	12	Double	Clad midpoint temperature at ZClad (K or C)
15	13	Double	Clad melt? (No - 0.0, Yes - 1.0)
16	14	Double	Coolant temperature at ZCool (K or C)
17	15	Double	Inner structure temperature at ZStruct (K or C)
18	16	Double	Channel outlet flowrate (kg/s per pin)
19	17	Double	Channel inlet flowrate (kg/s per pin)
20	18	Double	Channel outlet temperature (K or C)
21	19	Double	Channel inlet temperature (K or C)
22-25		Double(4)	Reserved
26		Integer	Reserved
27		Double	Reserved
28-29		Integer(2)	Reserved
Remaining		Double(3*KMAX)	Reserved

Table 2.5.2: Core-Wide Contents of CHANNEL.dat

Binary Index	CSV Column	Data Type	Description
1	1	Integer	Time step number (-)
2	2	Integer	Power option, **IPOWER** (-)
3	3	Double	Time (s). Note the CSV converter copies the time entry into column one of each channel specific file.
4	4	Double	Time step size (s)
5	5	Double	Normalized total power (-)
6	6	Double	Reactor period (s)
7	7	Double	Integrated normalized total power (-)
8	8	Double	Normalized decay power (-)
9	9	Double	Net reactivity ($)
10	10	Double	Programmed reactivity ($)
11	11	Double	Doppler reactivity ($)
12	12	Double	Axial expansion reactivity ($)
13	13	Double	Radial expansion reactivity ($)
14	14	Double	CRDL expansion reactivity ($)
15	15	Double	Coolant void reactivity ($)
16	16	Double	Fuel relocation reactivity ($)
17	17	Double	Clad relocation reactivity ($)
Remaining		Double(3* **NCHAN** + 1)	Reserved

2.5.2.2 PRIMAR4.dat

PRIMAR4.dat is output if requested by the user (**NBINOT** > 0). Data is written to PRIMAR4.dat at a frequency specified by the user (**INULLT**, **IBINST**). The PRIMAR4.dat file is an unstructured binary file. Converter utilities are provided as part of the distribution package. The converter utilities translate the binary data into a CSV or XML format and print the formatted data to the standard output.

Each time data is written to PRIMAR4.dat, a row is created that contains the values shown in Table 2.5.3. For non-restart cases, the first row in the PRIMAR4.dat file has the **IBINOT** information appended to the row. In this case, at binary index 5 + NCols **NBINOT** is written as an Integer. In columns (6 + NCols) - (5 + NCols + NBINOT) a transpose of the **IBINOT** information is written as an Integer. This information is not included in the binary file for restart cases and is therefore not utilized by the converter utility.

Table 2.5.3: Initial Contents of PRIMAR4.dat

Binary Index	CSV Column	Data Type	Description
1	1	Integer	Time step number
2	2	Integer	PRIMAR-4 time step number
3	3	Integer	NCols - Total number of output columns
4	4	Double	Time (s)
5 - (4 + NCols)	5 - (4 + NCols)	Double(NCols)	Output data requested in **IBINOT**

2.5.2.3 CONTROL.dat

CONTROL.dat is output when the control system is in use and the user requests the data is printed. Users can define the frequency of CONTROL.dat print using the *J1* parameter for signal 999. The CONTROL.dat file is an unstructured binary file. Converter utilities are provided as part of the distribution package. The converter utilities translate the binary data into a CSV or XML format and print the formatted data to the standard output.

CONTROL.dat contains a header section which defines the number of signals contained within the file and a description of the signals. The format of the header section is described in Table 2.5.4. Each time data is written to CONTROL.dat, *i*, two rows are created that contain the values shown in Table 2.5.5.

Table 2.5.4: Initial Contents of CONTROL.dat

Binary Entry	Binary Index	Data Type	Description
1	1	Integer	NSigs - total number of signals
2	1 - (NSigs - 1)	Integer(NSigs)	Signal Number
3	1 - (NSigs - 1)	Integer(NSigs)	Signal Type Id
4	1 - (NSigs - 1)	Character(32*NSigs)	Signal Type Description

Table 2.5.5: Main Contents of CONTROL.dat

Binary Entry	Binary Index	CSV Column	Data Type	Description
i	1	1	Integer	PRIMAR-4 time step number (-)
i	2	2	Double	Time at the beginning of the time step (s)
i + 1	1 - (NSigs - 1)	3 - (2 + NSigs)	Double(NSigs)	Signal values

2.5.2.4 STRATCV.dat

STRATCV.dat is output when the stratified volume model is active and the user requests the data is printed. Users can define the frequency of STRATCV.dat print using the **IFT16** parameter. The STRATCV.dat file is an unstructured binary file. Converter utilities are provided as part of the distribution package. The converter utilities translate the binary data into a CSV or XML format.

Each time data is written to STRATCV.dat, *i*, several rows are created that contain the values shown in Table 2.5.6 and Table 2.5.7. If users define a negative value for **IFT16**, additional debug information is included in the binary file, this information is presented in Table 2.5.8. Table 2.5.7, and Table 2.5.8 if debug information is requested, is written to the binary file for each stratified volume, *j*.

Table 2.5.6: Contents of STRATCV.dat

Binary Entry	Binary Index	CSV Column	Data Type	Description
i	1		Integer	Print frequency, **IFT16** (-)
i	2		Integer	NSTRCV - Number of stratified CVs (-)
i	3	1	Integer	PRIMAR-4 time step number (-)
i	4	2	Double	Time (s)
i + 1			Multi-line entry for each CV	see Table 2.5.7

Table 2.5.7: Contents of STRATCV.dat for a given stratified volume

Binary Entry	Binary Index	CSV Column	Data Type	Description
j	1	Appears in CSV file name	Integer	ICVSTR - Stratified CV number
j	2	3	Integer	NL4 - Number of layers
j	3	4	Integer	ISTAG4 - Stage number
j + 1	1	5	Double	ZLAYR4(1) - Bottom elevation of first layer (m)
j + 1	2	6	Double	TSTCV4(1) - Temperature of first layer (K)
j + 1	3	7	Double	XLSTM4(1) - Mass of first layer (kg)
j + 2	1	8	Double	ZLAYR4(2) - Bottom elevation of the second layer (m)
j + 2	2	9	Double	TSTCV4(2) - Temperature of second layer (K)
j + 2	3	10	Double	XLSTM4(2) - Mass of second layer (kg)
j + 3	1	11	Double	ZLAYR4(3) - Bottom elevation of third layer (m)
j + 3	2	12	Double	TSTCV4(3) - Temperature of third layer (K)
j + 3	3	13	Double	XLSTM4(3) - Mass of third layer (kg)
j + 4	1	14	Double	ZLAYR4(1+NL4) - Upper elevation of the last layer (m)

Table 2.5.8: Optional debug contents of STRATCV.dat for a given stratified volume

Binary Entry	Binary Index	CSV Column	Data Type	Description
j + 5	1	15	Double	WPLUME - Flow rate into plume (kg/s)
j + 5	2	16	Double	TPLUME - Temperature of plume (K)
j + 5	3	17	Double	Z0 - Plume established height (m)
j + 5	4	18	Double	V0 - Plume exit velocity (m/s)
j + 5	5	19	Double	VJ - Plume velocity at interface (m/s)
j + 5	6	20	Double	FR - Froude number (-)
j + 5	7	21	Double	HJET - Jet height (m)
j + 5	8	22	Double	WENTR - Flow entrainment rate (kg/s)

2.5.2.5 MFUEL Binary Files

As described in Section 9.6, when the MFUEL model is activated, a pre-transient and a transient binary file are created which contain the peak values of relevant parameters at the end of a main MFUEL time step. Two unique binary files are created for each channel, one for the pre-transient results and one for the transient results. A converter utility is provided as part of the distribution package. The converter utility translates the binary data into a CSV.

The MFUEL Peak value file contains a header section which defines the channel number, axial and radial meshing, the number of data columns and the time unit. The format of the header section is described in Table 2.5.9. Each time data is written to a peak axial file, *i*, a row is created that contains the values shown in Table 2.5.10.

MFUEL also outputs a pre-transient binary file including axial distributions of relevant parameters. Users can control the frequency at which the axial data is written during pre-transient characterization using the table-based **MSTEP** input. A unique binary file is created for each channel that uses MFUEL. A converter utility is provided as part of the distribution package. The converter utility translates the binary data into a multiple CSV files.

The axial-dependent MFUEL file contains a header section which defines the channel number, the number of included time steps, the axial meshing, and the number of data columns. The format of the header section is described in Table 2.5.11. Each time data is written to the axial-dependent file, *i*, *MZ* rows are created that contain the values shown in Table 2.5.12.

Table 2.5.9: Initial Contents of the Peak Value MFUEL Files

Binary Entry	Binary Index	Data Type	Description
1	1	Integer	iChan - channel number
1	2	Integer	MZ - number of axial fuel nodes
1	3	Integer	NEP - number of radial nodes
1	4	Integer	nCol - number of data columns
1	5	Character(3)	Unit of time, sec - seconds, or day - days

Table 2.5.10: Contents of MFUEL Peak Channel Files

Binary Entry	Binary/CSV Index	Data Type	Description
i	1	Double	Time (days for steady state file, seconds for transient file)
i	2	Double	Maximum burnup (at%)
i	3	Double	Maximum linear heat rate (kW/m)
i	4	Double	Maximum fuel temperature (K)
i	5	Double	Maximum clad inner temperature (K)
i	6	Double	Maximum clad hoop stress (MPa)
i	7	Double	Maximum fuel permanent axial expansion (%)
i	8	Double	Maximum fuel total axial expansion (%)
i	9	Double	Maximum clad permanent axial expansion (%)
i	10	Double	Maximum clad total axial expansion (%)
i	11	Double	Maximum fission gas release (%)
i	12	Double	Maximum plenum pressure (MPa)
i	13	Double	Maximum total clad hoop strain (%)
i	14	Double	Maximum total creep hoop strain (%)
i	15	Double	Maximum thermal creep hoop strain (%)
i	16	Double	Maximum irradiation creep hoop strain (%)
i	17	Double	Maximum volumetric swelling (%)
i	18	Double	Maximum total clad wastage (microns)
i	19	Double	Maximum lanthanide wastage (microns)
i	20	Double	Maximum eutectic wastage (microns)
i	21	Double	Maximum Na corrosion thickness (microns)
i	22	Double	Maximum CDF (-)
i	23	Double	Maximum MCF (-)
i	24	Double	Maximum neutron fluence (n/cm ^{2})
i	25	Double	Maximum clad dose (dpa)

Table 2.5.11: Initial Contents of the MFUEL Axial Files

Binary Entry	Binary Index	Data Type	Description
1	1	Integer	iChan - channel number
1	2	Integer	nSteps - number of time steps included in the file
1	3	Integer	MZ - number of axial nodes
1	4	Integer	nCol - number of data columns

Table 2.5.12: Contents of MFUEL Axial Files

Binary Index	Data Type	Description
1	Double	Axial location (m)
2	Double	Relative axial location (-)
3	Double	Cold cladding strain (%)
4	Double	Clad inner wastage (microns)
5	Double	Clad inner temperature (C)
6	Double	Clad hoop stress (MPa)
7	Double	Fuel outer radius (m)

2.5.3 Auxiliary Input and Output Files

The input and output files used in SAS4A/SASSYS-1 are listed in Table 2.5.13.

Table 2.5.13: SAS4A/SASSYS-1 Assigned External Files

File	Type	Assigned Use
stdin	Formatted	Input Data Records
stdout	Formatted	Printed Output
INPUT.dat	Formatted	Edited Input Data Records
8	Formatted	Scratch BOP Input Data
SAS.log	Formatted	Printed Output
CHANNEL.dat	Binary	Main Time Step Plotting
12	Binary	PLUTO2/LEVITATE Data for Plotting
13	Binary	PLUTO2/LEVITATE Data for Plotting
14	Binary	PLUTO2/LEVITATE Data for Plotting
PRIMAR4.dat	Binary	PRIMAR-4 Data for Plotting
CONTROL.dat	Binary	Control System Signal Data
16	Formatted	PINACLE/LEVITATE Data for Plotting
RESTART.dat	Binary	Output Restart File
RESTART.bin	Binary	Input Restart File
20	Binary	TSBOIL Data for Plotting
21	Binary	DEFORM-5 Plotting
22	Binary	EBR2 Reactivity Plotting
23	Binary	FPIN2 Data for Plotting
24	Binary	EBR-II Mark-V Safety Case Scratch Plotting Data File
26	Binary	BOP Data for Plotting
27	Binary	Steam Generator Data for Plotting
28	Binary	BOP Data for Plotting
29	Binary	BOP Data for Plotting
MaxTemps.txt	Formatted	Summary of Maximum Temperatures in the Core
MFUELss_ICH.dat	Binary	Summary of the MFUEL pre-transient irradiation predictions. Unique file for each channe
MFUELss_ICH-axial.dat	Binary	Axial summary of select MFUEL pre-transient irradiation predictions. Unique file for eac
MFUELts_ICH.dat	Binary	Summary of the MFUEL transient predictions. Unique file for each channel using MFUEL

2.6 Standard Input Description

SAS input data is read from a simple text file that is provided on the command line. The input file is normally provided via input redirection:

```
$ sas.x < model.inp > model.out
```

where `model.inp` is the model input file, `model.out` is the intended output file, and the executable, `sas.x`, depends on the version and platform being used.

Starting with Version 5.3, command-line arguments may also be used. The input and output files may be specified as the first and second positional arguments:

```
$ sas.x model.inp model.out
```

or they may be specified using command-line options:

```
$ sas.x --input model.inp --output model.out
```

Short versions of the command-line options (`-i` and `-o` respectively) may also be used.

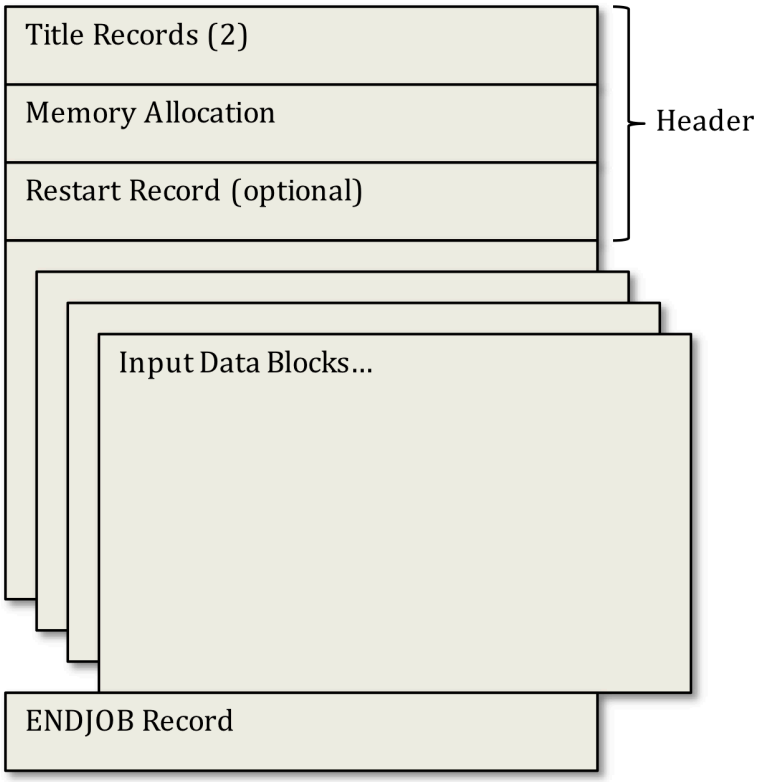
An additional command line option is available to override the default port to which SAS binds when listening for incoming connections to coupling interfaces. The default port is 60439, but this can be changed with the following:

```
$ sas.x --port <port>
```

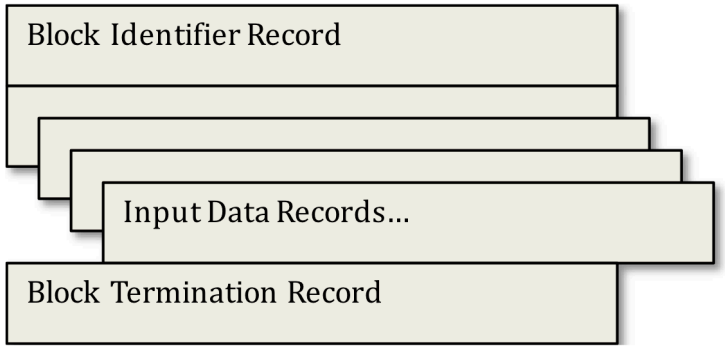
This type of interface is used, for example, with the *coupled RVACS model*. Instead of defining the port on the command line, SAS will also recognize the environment variable `__SAS_ENV_BIND_PORT__`. If both the environment variable and the command-line option are used, the command-line option takes priority.

2.6.1 Input Format

A SAS model file has the following structure:



where the Input Data Blocks have the following structure:



With the exception of the title records, any text following a comment character (! or #) will be stripped from the input before being parsed. When searching for the first title record, if any non-blank text exists before a comment character, then the comment is considered to be part of the title. The second title record must immediately follow the first.

2.6.2 Input Data Records Descriptions

Each input file for SAS4A/SASSYS-1 begins with a file header that includes two title records (primary and secondary) along with memory allocation information and an optional restart record. The first 72 characters of the primary and secondary title records will appear as a heading at the top of each page break in the printed output. Both title records are required.

Primary Title Record

```
form = "(A72,A8)"
read(*,form) TITLE(1), VERNAM
```

Input	Description
TITLE(1)	Primary Title (72 characters)
VERNAM	The major and minor code version label for this input deck. For Version X.y.z, the version label is the eight character string `X.ybbbbb` left adjusted in columns 73 through 80, where "b" is a blank character (space). The version label must match the current code version in order for execution to proceed beyond the reading of the primary title record.

Secondary Title Record

```
form = "(A72)"
read(*,form) TITLE(2)
```

Input	Description
TITLE(2)	Secondary Title (72 characters)

Storage Allocation Record

```
form = "(I6,9I3)"
read(*,form) NCH, NEUTSP, IDBUGP, IPDECK, NBYSSH, &
             IDATMO, IADEFC, IAPLUC, IACNTL, IALBOP
```

Input	Description
NCH	Number of channels for this case (see NCHAN). Maximum value is 22000.
NEUTSP	Neutronics storage allocation flag. Set > 0 when spatial kinetics modules are used. Only relevant with distributions that include coupled spatial kinetics capabilities.
IDBUGP	Data management print flag. Set > 0 when data storage allocation and data movement diagnostic prints are desired.
IPDECK	Input data editing flag. Set > 0 to generate a compiled copy (i.e. redefinitions eliminated, last defined values retained) of the input deck output as an ASCII file "INPUT.dat".
NBYSSH	Number of bypass channels involved in subassembly-to-subassembly heat transfer.
IDATMO	No longer used.
IADEFC	Data pack DEFC storage allocation flag. Set > 0 to eliminate storage allocation for data pack DEFC (DEFC is necessary for DEFORM-4 and DEFORM-5).
IAPLUC	Data pack PLUC storage allocation flag. Set > 0 to eliminate storage allocation for data pack PLUC (PLUC is necessary for PINACLE, LEVITATE, and/or PLUTO-2).
IACNTL	Control system module data storage allocation flag. Set > 0 to eliminate storage allocation for control system module data and input data block INCONT.
IALBOP	Balance-of-plant module data storage allocation flag. Set > 0 to eliminate storage allocation for balance-of plant module data and input data blocks IINBOP and FINBOP. When the balance of plant model is activated, steam generators do not participate in the null transient.

Neutronics Storage Allocation Record

The neutronics storage allocation record should be present only when `NEUTSP > 0`.

```
form = "(2I12)"
read(*,form) MAXFCM, MAXECM
```

Input	Description
MAXFCM	Number of single precision words in data buffer array.
MAXECM	Number of single precision words in data storage array.

RESTART Directive

The RESTART directive is required only for cases that initiate with the reading of a restart file from binary data file "RESTART.bin".

```
form = "(A8, I4, I6)"
read(*,form) NAMBLK, NUMBLK, IRN
```

Input	Description
NAMBLK	"RESTART" (left justified)
NUMBLK	100
IRN	Restart number on the restart file that this case will use. If IRN = 999, the restart used will be the last restart written on the restart file. If IRN = 0, the first restart written on the restart file will be used. (IRN > 0 is used only for the multiple restart option. See **MULSTR**.)

INCLUDE Directive

Note: New in Version 5.3.

Syntax:

```
INCLUDE <file>
```

An INCLUDE directive instructs the SAS input processor to read additional data from an external file. Once SAS has completed reading the external file, input processing continues with the next line in the current file. Included files may, themselves, have INCLUDE directives, however recursively including a file that is currently being read will result in an error.

INCLUDE directives are only interpreted in the body of the input file. The directive may not appear before a RESTART directive or before the memory allocation record.

Included files may not contain title records, memory allocation records, or RESTART or ENDJOB directives. They may contain any other input fragment that, when taken into context with the surrounding input, produces a valid SAS model file.

SAS records a log of all input read in the SAS.log file. The log file has been updated to report both the file and line number for each line of input.

Comment Records

Syntax:

```
# comment 1
! comment 2
! etc...
```

A comment record is any input record whose first non-blank character is a ! or # symbol. Arbitrary text may follow the comment character. Comment records have an advantage over XNOTES input blocks in that they may occur anywhere within an input description, including between the input records of other input blocks.

Block Identifier Record

The block identifier record is required as the first record of each input block.

```
form = "(A8, I4, 2I6)"
read(*,form) NAMBLK, NUMBLK, ICH, IZERO
```

Input	Description
NAMBLK	Block name (left justified).
NUMBLK	Block number (see Table I).
ICH	Channel number (use 1 if NUMBLK < 51).
IZERO	0, Zero block background input before reading data. Do not use 0 for data input following the reading of a restart file. >0, Channel number of that block of the same name whose input data is to be used as background input before reading data. Background input blocks must be previously defined. Use IZERO = ICH for data input during a restart.

Block Input Data Records

Within each block, input data is read in a format consistent with the data type for that block: integer, floating point, or character. See the list of *Formatted Input Blocks* below for the data type associated with each input data block.

Table 2.6.1: Formatted Input Blocks

NUMBLK	NAMBLK	Input Format
1	INPCOM	Integer
3	INPMR4	Integer
5	INCONT	Special Mixed
6	IINBOP	Special Integer
7	INEUTR	Integer
11	OPCIN	Real
12	POWINA	Real
13	PMATCM	Real
14	PRIMIN	Real
15	FINBOP	Special Real
16	RNEUTR	Real
17	ANEUTR	Character
18	PMR4IN	Real
51	INPCHN	Integer
61	GEOMIN	Real
62	POWINC	Real
63	PMATCH	Real
64	COOLIN	Real
65	FUELIN	Real

Fixed Format Integer Input

```
form = "(2I6,10I6)"
read(*,form) LOC, NUM, IDATA(LOC:LOC+NUM-1)
```

Input	Description
LOC	Initial block location for data input.
NUM	Number of input data to be read into the IDATA fields (1 <= NUM <= 10).
IDATA	Input data definitions to be placed in locations (LOC) to (LOC+NUM-1) in the input block.

Fixed Format Floating Point Input

```
form = "(2I6,5F12.0)"
read(*,form) LOC, NUM, RDATA(LOC:LOC+NUM-1)
```

Input	Description
LOC	Initial block location for data input.
NUM	Number of input data to be read into the RDATA fields (1 <= NUM <= 5).
RDATA	Input data definitions to be placed in locations (LOC) to (LOC+NUM-1) in the input block.

Fixed Format Character Input

```
form = "(2I6,10A6)"
read(*,form) LOC, NUM, CDATA(LOC:LOC+NUM-1)
```

Input	Description
LOC	Initial block location for data input.
NUM	Number of input data to be read into the CDATA fields (1 <= NUM <= 10).
CDATA	Input data definitions to be placed in locations (LOC) to (LOC+NUM-1) in the input block.

Free-Formatted Input *(New in Version 5.3)*

Syntax:

```
<LOC> : <DATA> [<DATA>...]
```

Input	Description
LOC	Starting block location for data input.
DATA	Input data definitions to be placed in the input block beginning at location (LOC) and continuing until all input is consumed on the line.

Free formatted input may be used for integer or floating point data blocks. Free-format lines require the specification of the initial location, LOC followed by a colon (:) and any number of space-, comma-, or tab-delimited

input values in sequential location order. The initial location value may be indented with leading white space. Optional whitespace may be on either side of the colon, which is required. At least one numeric value must be present after the colon.

Consecutive spaces and tabs are treated as a single white space. However consecutive commas are significant and cause the corresponding input location to be skipped. Comments may follow the data.

Input blocks may contain both fixed- and free-formatted records, however the data type must be compatible with the block type. Listing 2.6.1 provides an example of free-formatted input.

Listing 2.6.1: Example of Acceptable Free-Format Input

```
#---+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8
#
#  INPCOM: Channel Independent Variables (Integer)
#
INPCOM     1     0     0
#
#        MAXSTP: Maximum Number of Main (Power and Reactivity)
#        | Time Steps
    11 : 0                                          !MAXSTP
#
#        IPO: Number of Steps between Prints before IBLPRT or Boiling
#        |     IPOBOI: Number of Steps between Prints after IBLPRT
#        |     |       or Start of Boiling
#        |     |
    12 : 1000 1000                                  #IPO, IPOBOI
#
#      NOREAC: Main Time Step Intevals between PSHORT print
#      |
    41:1000
#
#       ICLCMP: Flag to Save Plot Data for Transients to Unit 11
#       |
24:     1                                           !ICLCMP
    -1
#---+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8
```

Block Termination Record

Syntax:

```
END
```

The block termination record is required as the final record of each input block.

As an alternative to the `END` statement, SAS still recognizes a fixed-formatted input record where `LOC = -1` as a block termination record.

ENDJOB Directive

The ENDJOB directive indicates the final line of input. Input processing terminates when this record is encountered.

```
form = "(A8, I4)"
read(*,form) NAMBLK, NUMBLK
```

Input	Description
NAMBLK	"ENDJOB" (left justified)
NUMBLK	-1

XNOTES Block

Format:

```
form = (A72)
read(*,form) COMMENT
```

Comments can be added in the input file as a pseudo input block called XNOTES. The block begins with a block identifier record having the block name 'XNOTES', followed by any number of comment lines. The block ends with an END statement or a -1 in columns 5-6. The comment records are not stored and are not transferred to the edited input data file INPUT.dat or the restart file RESTART.dat.

Example:

```
XNOTES ! (block identifier)
.
.
.
(COMMENTS)
.
.
.
END ! (block delimiter)
```

TABLE Block

Table input blocks allow users to define tables of values that can be referenced by other input and models.

General syntax:

```
TABLE <id> <name>
      col_1      col_2      col_3      [...] ! column labels
    ! val(1,1)   val(1,2)   val(1,3)   [...] ! row 1
    ! val(2,1)   val(2,2)   val(2,3)   [...] ! row 2
    ! [...]                                  ! row n
END
```

Table blocks are identified by the keyword TABLE followed by a unique *id* and *name*. Table blocks are terminated by an END statement. The column labels define the number of columns in the table. The labels required will depend on the table application. Some applications have specific requirements for the column labels.

Stand-alone tables in the input file have global scope and are accessible by all models. Tables may be defined within certain other blocks, including *FUNCTION*, *EMPUMP*, or *CHANNEL* blocks. A table defined within another block has local scope inside that block and is only accessible by the enclosing model or channel.

FUNCTION Block

Function input blocks allow users to construct arbitrary functions that can be referenced by other input and models. Five types of user functions can be defined: table lookup (interpolated), field, user-defined (parsed), external dynamic library, and control system.

General syntax:

```
FUNCTION <id> <name>
    TYPE = <type>
    ! [...]
END
```

Function blocks are identified by the keyword FUNCTION followed by a unique *id* and *name*. Function blocks are terminated by an END statement. The *type* parameter is required, while the presence of other parameters depends on the type. Five values correspond to the five types of user functions that are presently supported: TABLE, FIELD, USER, EXTERNAL, and CONTROL.

The remaining parameters that can be used are *source*, *func*, *arg*, and *kind*. Interpretation of these parameters depends on the type of the function being defined. In addition, function blocks may contain table blocks, but these are only relevant for table lookup functions.

As examples, functional inputs may be used to specify transient external reactivity or power (see **NPREAT**), coolant inlet temperature for PRIMAR-1 (see **NT0TAB**), coolant driving pressure or normalized coolant flow rate for PRIMAR-1 (see **NPRES**), or ALMR EM Pump trip time (see **ILRPMP**). Additional uses for function blocks are described throughout the manual.

Internally, arguments are passed to function blocks as elements of a one-dimensional array, and each function returns a scalar result. The number of arguments passed to the function block depends on the model invoking the function. Therefore the interface is implicit, and it is up to the user to provide a meaningful function block. The function block defines how, or even if, the arguments are used.

As an example, a function block can be used to represent power or reactivity, similar to how the **PREATB** and **PREATM** input works, and it will be passed the current simulation time. The function block is expected to use the argument to determine the current power or reactivity. A TABLE function block will use the argument to interpolate over a table of values. However, if a CONTROL function block is used in this case, it will ignore the argument and simply return the current value of the specified control signal.

Table Lookup Functions:

Table lookup functions can be used to define a variety of interpolations based on a user-supplied data table. For example, linear interpolation of the first four squares can be accomplished with the following function block:

```
FUNCTION 4 Squares
    TYPE = TABLE
    TABLE 2 Data
        x  y
        0  0
        1  1
        2  4
        3  9
        4  16
```

(continues on next page)

(continued from previous page)

```
    END
END
```

Table lookup functions support four parameters: *source*, *func*, *arg*, and *kind*. The four parameters are all optional. Interpretation of these parameters is shown in Table 2.6.2.

Table 2.6.2: Table Lookup Function Block Parameters

Parameter	Description	Default
source	Table *id* of local data source to be used for interpolation	First table found in function
arg	Column label of independent variable in the data source table	First column
func	Column label of dependent variable in the data source table	Second column
kind	Type of interpolation to be performed	Linear

Using these parameters, users can provide more complex data tables and have more control over the type of interpolation that will be performed. Extending the previous example, the following function will perform natural spline interpolation of the first four squares:

```
FUNCTION 4 Squares
    TYPE = TABLE
    SOURCE = 2
    ARG = x
    FUNC = y2
    KIND = Spline
    TABLE 2 Data
        row x   y1  y2
        1   0    0   0
        2   1    1   1
        3   2    2   4
        4   3    3   9
        5   4    4  16
    END
END
```

The available types of interpolation are summarized in Table 2.6.3. In all cases, interpolating values that occur before the first (or after the last) argument in the data table will use the first (or last) function value. No extrapolation is performed. Source data tables may be of arbitrary length, limited only by available memory.

Table 2.6.3: Interpolation Options for Table Lookup Functions

Interpolation *KIND*	Description
Linear	Linear interpolation of successive values in the source data table.
LogLinear	Linear interpolation of the natural logarithm of successive values in the source data table. The returned value is the exponential of the interpolated value.
Spline	Natural (cubic) spline fit that enforces continuity in the value and first and second derivatives at each data point. For closure, a natural spline defines the second derivative to be zero for the first and last data points.
LogSpline	Natural spline fit that enforces continuity in the logarithm of the value and first and second derivatives at each data point.
HermiteL0	Quadratic fit based on second-order Hermite polynomials that enforces continuity of the value and first derivative at each data point. The slope at each interior point is determined by a weighted average of the slopes of the two adjacent intervals. For closure, the slopes at the end points are set to zero.
HermiteL1	Same as HermiteL0, except that the slopes at the first and last points are taken as the slopes of the first and last intervals, respectively.
HermiteQ0	Quadratic fit based on second-order Hermite polynomials that enforces continuity of the value and first derivative at each data point. The slope at each interior point is determined by a parabolic fit with the two adjacent data points. For closure, the slopes at the end points are set to zero.
HermiteQ1	Same as HermiteQ0, except that the slopes at the first and last points are taken as the slopes of the first and last intervals, respectively.
HermiteQ2	Same as HermiteQ0, except that the slopes at the first and last points are taken as the slope of the adjacent interpolating parabola.

Field Functions:

Field functions can be used to perform two-dimensional interpolation over a tabulated field of data. For example, bilinear interpolation of the sum of two values can be accomplished with the following function block:

```
FUNCTION 8 Sum
    TYPE = FIELD
    TABLE 1 FieldData
          x\y 0  1  2  3  4
           0  0  1  2  3  4
           1  1  2  3  4  5
           2  2  3  4  5  6
           3  3  4  5  6  7
           4  4  5  6  7  8
    END
END
```

Field functions support two parameters: *source*, and *kind*. Both parameters are optional. If *source* is not provided, the first table contained in the function will be used. Currently only bilinear interpolation is supported for field functions. If *kind* is not specified, it will default to the only option, BiLinear.

Field functions use the entire *source* table as the data field. The first column in the data table is interpreted as the coordinate values for the first dimension. The second and subsequent *column labels* in the TABLE block are interpreted as coordinate values for the second dimension. The first label is required but ignored.

Given the following source table:

$$\begin{array}{cccccccc} < any > & y_0 & y_1 & \dots & y_j & \dots & y_J \\ x_0 & f_{0,0} & f_{0,1} & & & & \\ x_1 & f_{1,0} & f_{1,1} & & & & \\ \dots & & & \dots & \dots & & \\ x_i & & & \dots & f_{i,j} & & f_{i,J} \\ \dots & & & & & \dots & \dots \\ x_I & & & & f_{I,j} & \dots & f_{I,J} \end{array}$$

where $f_{i,j} = f(x_i, y_j)$ is the function value at the coordinate x_i, y_j, bilinear interpolation can be performed for any x, y coordinate pair. If

$$x_{i-1} < x \leq x_i$$
$$y_{j-1} < y \leq y_j$$

then

$$f(x,y) \approx (f_j(x) - f_{j-1}(x)) \frac{y - y_{j-1}}{y_j - y_{j-1}} + f_{j-1}(x)$$

where

$$f_{j-1}(x) = (f_{i,j-1} - f_{i-1,j-1}) \frac{x - x_{i-1}}{x_i - x_{i-1}} + f_{i-1,j-1}$$
$$f_j(x) = (f_{i,j} - f_{i-1,j}) \frac{x - x_{i-1}}{x_i - x_{i-1}} + f_{i-1,j}$$

If either coordinate falls outside the range of the provided coordinates, flat, constant extrapolation is performed. More specifically:

$$x < x_0 \text{ and } y < y_0 : f(x,y) \approx f(x_0, y_0)$$
$$x > x_I \text{ and } y < y_0 : f(x,y) \approx f(x_I, y_0)$$
$$x < x_0 \text{ and } y > y_J : f(x,y) \approx f(x_0, y_J)$$
$$x > x_I \text{ and } y > y_J : f(x,y) \approx f(x_I, y_J)$$
$$x < x_0 \text{ and } y_0 < y < y_J : f(x,y) \approx f(x_0, y)$$
$$x > x_I \text{ and } y_0 < y < y_J : f(x,y) \approx f(x_I, y)$$
$$x_0 < x < x_I \text{ and } y < y_0 : f(x,y) \approx f(x, y_0)$$
$$x_0 < x < x_I \text{ and } y > y_J : f(x,y) \approx f(x, y_J)$$

User-Defined Functions:

User-defined functions support a basic syntax for building complex expressions from mathematical operators and functions. For example, a sinusoidal function can be defined with the following function block:

```
FUNCTION 3 Sinus
    TYPE = USER
    ARG = t
    FUNC = "pi := 3.14159; 2.0*sin(pi*t)"
END
```

The two parameters arg and func are required for user functions. Values for source and kind are ignored.

Parsing and evaluation of user-defined functions is handled by the open-source fparser library [2-10]. As shown in the example above, the function syntax supports inline variables and numeric literals. Logical expressions are also supported. For example,

```
FUNCTION 2 Squares
    TYPE = USER
    ARG = x
    FUNC = "if(x>0,x^2,0)"
END
```

will return the square of positive values of x, and zero otherwise. For details on the supported syntax, see Reference [2-11].

External Dynamic Functions:

Compiled functions contained in external dynamic libraries (dylib on macOS, DLL on Windows, and shared objects on Linux) can be accessed via an external function. For example, given a dynamic library MyLib.dylib that contains the function MyFunc, the function block would be defined as follows:

```
FUNCTION 7 ExternLib
    TYPE = EXTERNAL
    SOURCE = MyLib.dylib
    FUNC = MyFunc
END
```

With this definition, SAS4A/SASSYS-1 will dynamically load "MyLib.dylib" and link to the function "MyFunc". The source and func parameters are required. Other parameters are ignored.

Control System Functions:

The Control System module can react to and interact with many plant state variables. To support interaction with the plant state outside the Control System, a control function block can be defined. This function block requires that a control system model be defined by an INCONT block elsewhere in the input. A control function has the following simple form:

```
FUNCTION <id> <name>
    TYPE = CONTROL
    FUNC = <signal> # e.g. FUNC = 23
END
```

The single parameter func defines which signal number to access in the Control System model. This signal number may come from a demand table or may be the output from a block signal in the control system logic. Arguments to this type of function block are ignored.

EMPUMP Block

Model data for *Equivalent Circuit EM pumps* is specified using EMPUMP blocks. The syntax of EMPUMP blocks is discussed here, while the specific input data requirements for each EM pump model is discussed in *Chapter 5*.

General syntax:

```
EMPUMP <id> <type> [<name>]

    key_1 = value_1
    key_2 = value_2
    ! [...]

    ! FUNCTION blocks...
```

(continues on next page)

(continued from previous page)

```
    ! TABLE blocks...

END
```

Each EMPUMP block is denoted by the block identifier line, itself containing the case-sensitive `EMPUMP` specifier, a unique integer block *<id>*, the model *<type>* (`SIMPLE` or `DETAILED`, case-sensitive), and an optional block *<name>*. The block internals include required and optional keyword/value entries, in addition to optional *FUNCTION* and *TABLE* block entries, followed by the keyword END to terminate the block.

Rules of the EMPUMP block are as follows:

- Keyword entries are case-insensitive and may be defined in any order
- Separate keyword/value entries may be separated by a newline character, a comma, or whitespace within a line
- Whitespace between the keyword, value, and "=" sign is ignored
- FUNCTION and TABLE blocks may be defined within the EMPUMP block, in which case they have local scope, or outside the EMPUMP block as standalone blocks, in which case they have global scope
- Blocks defined within the EMPUMP block may come before or after any of the keyword/value entries
- Multiple parameters may call to the same within-scope FUNCTION block
- All required keyword/value entries for a given pump must be specified within a single EMPUMP block
- Parameters provided multiple times within a block take the value specified closest to the end of the block
- Comments may be placed in EMPUMP blocks using either the "!" or "#" characters, after which any characters on the line are ignored

At the moment, the model *type* may be either "SIMPLE" or "DETAILED", corresponding to the *simple* and *detailed* equivalent circuit models, respectively.

CHANNEL Block

A CHANNEL block is a container for any number of channel-dependent *TABLE* and *FUNCTION* blocks.

General syntax:

```
CHANNEL <ich>

    TABLE <id> <name>
        col_1 col_2 ...
        ! [...]
    END


    FUNCTION <id> <name>
        ! [...]
    END

    ! [...]

END
```

CHANNEL blocks are identified by the keyword `CHANNEL` followed by the associated channel number. Each CHANNEL block must be terminated by an `END` statement. The order of the *TABLE* and *FUNCTION* block definitions does not matter. When *TABLE* and *FUNCTION* blocks are defined inside a CHANNEL block, their scope is limited to models for that channel only. When channel-dependent models query Table or Function IDs, the search begins within the channel's scope. If the associated ID is not found, the search continues in the global scope.

Within the scope of each channel, Table and Function IDs, respectively, must be unique. A channel-dependent Table or Funcion ID will hide any global Table or Function with the same ID. This allows, for example, a global Table or Function ID to be used for most channels, while creating an alternate Table or Function with the same ID within certain CHANNEL blocks to override the global definition.

2.7 Input Data Blocks

2.7.1 Block 1 — INPCOM — Channel-Independent Options and Integer Input

1 **NCHAN**

Number of core channels. Maximum = 22000. (not to be confused with the bypass channels of PRIMAR-4). **NCHAN** must be less than or equal to NCH from the data allocation record.

2 **IDBUG0**

Channel independent debug flag.

= 0, No debugging prints.
> 0, Steady-state debug prints and transient time step prints.
= 2, Steady state coolant debug prints.
= 3, More steady state coolant debug prints.

3 **IFUEL1**

Number of fuel types (Max. = 8).

Note that the MFUEL model (**IFUELO** = 2) does not use the fuel types defined by the PMATCM input block.

4 **ICLAD1**

Number of cladding types (Max. = 3).

5 **IPLUP**

Fission gas plenum location:

= 0, Fuel-pin gas plenum above the core.

= 1, Fuel-pin gas plenum below the core.

Note that **IPLUP**= 0 is required for the MFUEL model (**IFUELO** = 2).

6 **IPROPT**

DEFORM-4 printed output selection option.

= 0, Use time steps to control printing.
= 1, Use absolute times based on **PRTSTR** and **PRTDEL** as well as time steps to control printing.

7 **ITKEL**

Temperature scale for SSPRNT and TSPRNT print routines and TSPLOT plotting routine.

= 0, Temperatures in Kelvins.
> 0, Temperatures in degrees centigrade.

8 **IPOWER**

Power driver option.

= 0, External reactivity vs. time from PREA subroutine or table. Total reactivity is the sum of the external reactivity and the internally-calculated feedback reactivity.
= 1, Power vs. time from PREA subroutine or table.

9 **IPOWOP**

Power option.

= 0, Steady-state power in the peak axial segment is calculated from total reactor power (See **POWTOT**).
> 0, Steady-state total reactor power is calculated from the peak axial segment . (See **POW**).

10 **NPK**

Reactor neutronics option.

= 0, Point kinetics with input perturbation theory reactivity worth tables.

= 1, Point kinetics with DIF3D (finite difference) calculated perturbation theory reactivity worth tables.
= 2, DIF3D-K or VARIANT-K nodal space-time neutronics using direct solution.
= 3, DIF3D-K nodal point kinetics with first order perturbation theory reactivity feedbacks.
= 4, DIF3D-K nodal point kinetics with adiabatic reactivity feedbacks.
= 5, DIF3D-K or VARIANT-K nodal space-time neutronics using improved quasi-static method.

11 MAXSTP

Maximum number of main (power and reactivity) time steps in transient calculation.

= 0, Steady-state calculation only.
> 0, Maximum number of time steps in the transient calculation.
= -1, Do not limit the number of times steps in the transient calculation.

12 IPO

Number of main (power and reactivity) time steps between full printouts before **IBLPRT** or coolant boiling (See **IBLPRN**).

13 IPOBOI

Number of main (power and reactivity) time steps between full printouts after **IBLPRT** or coolant boiling.

14 IBLPRT

Main time step number for switch of full printout interval from **IPO** to **IPOBOI**. If **IBLPRT** = 0, switch occurs at coolant boiling inception.

15 NSTEP

Restart file save option. For any nonzero value of **NSTEP**, the restart file is saved at the end of the steady-state calculation and when the transient simulation ends with either the maximum problem time, **TIMAX**, or the maximum number of time steps, **MAXSTP**.

= 0, Do not save the restart file.
< 0, Only save the restart file following completion of the steady-state and transient simulations.
> 0, Save the restart file every **NSTEP** main time steps during the transient, in addition to saving the restart file following completion of the steady-state and transient simulations.

16 **NDELAY**

Number of delayed neutron precursor families,

$0 \leq$ **NDELAY** ≤ 6.

17 **NDKGRP**

Number of decay heat groups. $0 \leq$ **NDKGRP** ≤ 24

The number of decay heat groups defined for any given curve may not exceed **NDKGRP** (except as noted below for **NDKGRP** = 0) but may be less than **NDKGRP**. This allows standard curves to be combined with user-supplied curves with a lower expansion order.

Decay heat calculations can be disabled by setting **NDKGRP** = 0 even if other decay heat input is present.

18 **NPREAT**

Power or reactivity option

$0 \leq$ **NPREAT** ≤ 20, Number of entries in PREA vs. time table (See **IPOWER**, **PREATB**, and **PREATM**).

> 20, Function ID for function of power or reactivity (see **IPOWER**, *FUNCTION* block)

19 **NPRES**

Coolant driving pressure option.

= 0, The exponential decay of pump head is used (see **PDEC**, **PDEC1**, **PDEC2**).

$0 <$ **NPRES** ≤ 20, Number of entries in table of coolant driving pressure or normalized coolant flow rate vs. time (see **IFLOW**, **PRETAB**, **PRETME**).

> 20, Function ID for function of coolant driving pressure or normalized coolant flow rate (see **IFLOW**, *FUNCTION* block).

20 **IFLOW**

Used only if **IPRION** $\neq 4$, **NPRES** > 0. If **IFLOW** = 0 and **IPRION** $\neq 4$, then the coolant driving head is specified as a function of time. If **IFLOW** > 0, **IFLOW** is the channel number in which coolant flow is specified as a function of time.

21 **NONEU0**

Reserved for space-time options.

22 **NT0TAB**

Coolant inlet temperature

0 ≤ **NT0TAB** ≤ 20, Number of entries in **T0TAB** vs. **T0TME** table, i.e., coolant inlet temperature vs. transient time table.

> 20, Function ID for function of coolant inlet temperature (see *FUNCTION* block).

23 **MULSTR**

Multiple restart option.

= 0, Normal restart option: Restarts written at multiples of **NSTEP**. Only the latest restart file is saved.
= 1, Multiple restart option: Restarts written at multiples of **NSTEP**. All restart files are concatenated. Not recommended.

24 **ICLCMP**

Core channel plotting option.

= 0, Data for plots of transient not saved.
> 0, Data output to CHANNEL.dat for transient plotting. See also **MSTPLA**, **MSTPLB**, **MSTPL1**, **MSTPL2**, and **MSTPL3**

25 **INEDIT**

Input edit option.

= 0, No input edit.
> 0, Input edit to standard output.

26 **IYLD**

Cladding flow stress formulation.

= 0, Based on TID-26666
= 1, DiMelfi - Kramer strain, strain-rate, temperature, and fluence dependent model.
= 2, Same as 1, but with high strain rate approximation.
= 3, Input table **YLDTAB**.

27 **IPRION**

PRIMAR-4 option.

= 4, For PRIMAR-4 option.
≠ 4, For PRIMAR-1 option.

28 **IMELTV**

Axial extent assumptions for the molten cavity pressure calculation.

= 0, Extends only over axial extent of melting.
= 1, Cavity includes all central cavity.
= 2, Each axial node treated as separate cavity.

29 **INAS3D**

Alkali metals coolant properties. (Overridden by **ID2O**, **IPBDEN**, **KPROPI**, and **ICLPRP**).

= 0, For SAS4A sodium properties.
> 0, For SAS3D sodium properties.
< 0, For NaK properties.

30 **NCLADM**

Not currently used.

31 **ICREXP**

Control rod drive expansion option

= 0, No control rod drive expansion feedback.
= 1-3, Calculate feedback from single-node model. (Option 2 switches on CRD output in older decks, but use of **ICRDDB** flag to control output is recommended).
= 4, Calculate feedback from multinode model.

32 **IXSTPC**

Controls when the DEFORM computed axial expansion calculation is stopped based on cladding conditions.

= 0, Continues.
= 1, Inner cladding node temperature > **TESOL**(**ICLADV**).
= 2, Inner two cladding nodes temperature > **TESOL**(**ICLADV**).
= 3, All cladding nodes temperature > **TESOL**(**ICLADV**).

33 IXSTPF

Controls when the DEFORM computed axial expansion calculation is stopped based on fuel melt percentage molten.

= 0, Continues.
> 0, Percent molten (0 < **IXSTPF** <= 100).

34 IMCVTY

Controls how much crack volume is included in the volume of the molten cavity. See **IROR**.

35 IDBPWI

= 0, No debug print for POWING subroutine.
= 1, Debug print for POWING subroutine.

36 IRADEX

Radial expansion feedback option.

= 0, No radial expansion feedback.
> 0, Radial expansion feedback using the inlet coolant temperature.
< 0, Radial expansion feedback using the inlet plenum wall temperature.

Simple radial expansion model

=1,-1 Calculate radial expansion reactivity feedback only.
=2,-2 Print the radial expansion reactivity feedback, inlet temperature and average structure temperature every time step.
=3,-3 Print debug.

Detailed radial expansion model

=4,-4 Print the radial expansion reactivity feedback, inlet temperature and average structure temperature every time step.

=5,-5 Print axial core shape, net deflection, and radial expansion with curve, plus debug, every time step.

=6,-6 FFTF core restraint design. See ANL/RAS 88-6 for consistent set of data. See program listing for additional cautions.

=7,-7 FFTF core restraint design, print axial core shape, net deflection, and radial expansion with curve, plus debug, every time step.

37 KFAILP

= 0, Mechanistic axial pin failure propagation in PLUTO2 and LEVITATE. (See **FNARME**).

= 1, Axial pin failure propagation determined by input. (See **TEFAIL**, **FNARME**, and **PRFAIL**).

38 NCPLEV

Switch from PLUTO2 to LEVITATE when **NCPLEV** axial cladding nodes have exceeded the cladding liquidus temperature.

Suggested value: 3.

NCPLEV = 0: Do not switch from PLUTO2 to LEVITATE due to cladding melting.

A switch from PLUTO2 to LEVITATE occurs automatically if all fuel and cladding in the pin are molten in any axial node.

39 NFUELD

Number of dollars of fuel reactivity to terminate the calculation. Recommended value: -5. | | PL,LE |

40 IAREXT

Controls radial extent of integration when calculating the plain strain axial expansion.

= 0, Over the solid fuel annulus.

= 1, Over all the fuel, both solid and molten.

41 NOREAC

Reactivity print option.

= 0, Full output from PSHORT.

> 0, Full output from PSHORT every **NOREAC** main time steps.

42 **NSLEEX**

Number of fully molten hexcan cells in a subassembly to terminate the calculation.

43 **NSRMTB**

Not currently used.

44 **INRAEJ**

= 0, Parametric fuel injection calculation. (See **CIPINJ**)
= 1, Ejection of in-pin fuel is calculated using a mechanistic model.

45 **NPOWDK**

Number of power curves or sets of user-defined decay heat curves used. $0 \leq$ **NPOWDK** ≤ 5

For **IPOWER** = 0, **NPOWDK** defines the number of user-defined decay heat curves to expect in the input. (Do not count any built-in ANS standard curves that are used in the problem.) **NPOWDK** should be zero when using the original (Version 1.0) decay heat input.

For **IPOWER** = 1, **NPOWDK** defines the number of user-supplied power vs. time tables. This usage is unrelated to the decay heat model.

46 **NPDKST**

Number of entries in the **POWLVL** vs. **POWTIM** and **PWLVL2** vs. **PWTIM2** tables for performing steady-state initialization of the decay heat precursors.

$0 \leq$ **NPDKST** ≤ 8

If **NPDKST** = 0, infinite steady-state irradiation is assumed for all decay heat regions regardless of the input in **POWLVL**/**POWTIM** and **PWLVL2**/**PWTIM2**.

47 **ICHUNK**

= 0, The chunk model in LEVITATE is not used.
= 1, The chunk model in LEVITATE is used.

48 **ILUBLK**

= 0, Inhibit chunk formation due to bulk freezing of the fuel.
= 1, Allow chunk formation via bulk freezing if no solid support for chunk formation is present.

49 **INAPN**

Sodium vapor pressure treatment.

= 0, No sodium vapor pressure in the fuel pin cavity.
= 1, Sodium vapor pressure is considered in the fuel pin cavity.

50 **NOEQPN**

Controls the pressure in the molten cavity at the time of PINACLE initiation.

= 0, The pressure in the cavity is equal to the DEFORM calculated pressure. If the input **PCFAIL** > 1.0, the pressure is set equal to **PCFAIL**.
= 1, The pressure in the cavity is calculated using the available fission gas.

LOCATIONS 51-54 USED ONLY IF |IRADEX| > 3

51 **NSUBTC**

Total number of subassemblies in the active core region, including control and internal blanket subassemblies. Used for calculating the core radius in the radial expansion reactivity feedback calculation.

52 **MTGRD**

Support grid material, used for calculating the thermal expansion of the grid during a transient.

= 1, 316 SS.
= 2, HT-9.

53 **MTACLP**

Above-core load pad material. Used for calculating the thermal expansion of the above-core load pad during a transient.

= 1, 316 SS.
= 2, HT-9.

54 **MTTLP**

Top load pad material, used for calculating the thermal expansion of the top load pad during a transient.

= 1, 316 SS.
= 2, HT-9.

55 **MODEEX**

Axial expansion option.

= 0, Force balance or free expansion, depending on radial gap.
= 1, Cladding controlled fuel expansion.
= 2, Independent free expansion.
= 3, Force balance all the time.

56 JREEXT

= 0, No correction term.
= 1, Add correction term to reactivity extrapolation.

57 IFT19

Not currently Used.

58 IREACT

Reactivity feedback model option.

= 0, Use detailed reactivity models.
= 1, Use FFTF empirical reactivity model.
= 2, Use EBR-II empirical reactivity model.
= -1 Compute, write out but do not use FFTF model.
= -2 Compute, write out but do not use EBR-II model.

LOCATIONS 59-65 USED ONLY IF |IRADEX| > 3

59 NSUBTR

Total number of subassemblies in the reactor, including drivers, radial and internal blankets, control assemblies, radial reflectors and shields.

60 NRRNGS

Number of restraint rings in core restraint design.

= 0, No restraint rings.
= 1, Restraint ring at top load pad only.
= 2, Restraint rings at top and above-core load pads.

61 MTRRAC

Material of the restraint ring at the above-core load pad elevation.

= 1, 316 SS.
= 2, HT-9.

62 **MTRRT**

Material of the restraint ring at the top load pad elevation.

= 1, 316 SS.
= 2, HT-9.

63 **MTRFAC**

Radial reflector and/or blanket above-core load pad material.

= 1, 316 SS.
= 2, HT-9.

64 **MTRFT**

Radial reflector and/or blanket top load pad material.

= 1, 316 SS.
= 2, HT-9.

65 **IROPT**

Assumption for low power-to-flow ratios.

= 0, Subassemblies vertical at the grid plate.
= 1, Above-core load pads remain compacted (most pessimistic).

66 **JCRIND**

For use with EBR-II reactivity model only.

This is the lowest node in which the control rod is present in the special EBR-II control rod feedback model. Sums then run from `JCRIND` to the top of S/A.

67 **ID2O**

Heavy water coolant properties.

(Overridden by **IPBDEN**, **KPROPI**, and **ICLPRP**).

= 0, Na or NaK properties (See **INAS3D**).
= 1, D2O properties.

68 **IDNFLW**

= 0, For initial upflow through the core.
= 1, For initial downflow through the core.

69 **IPLTSG**

Number of main time steps between saving plot data for the steam generator model.

70 **IBOP**

Balance of plant modeling option.

= 0, If balance-of-plant model is not used.
= 1, If balance-of-plant model is used.

When the balance of plant model is activated, steam generators do not participate in the null transient.

71 **ICRDDB**

Control rod drive expansion print option.

= 0, No output from control rod drive expansion model.
= 1, Regular CRD output every main time step.
= 2, Additional debug output every PRIMAR time step.

72-74 **ICRTMP (K)**

Specifies the environment temperature for section K of the control rod drive for **ICREXP** = 2. If = 0, the UIS temperature is used. If > 0, the temperature of compressible volume **ICRTMP** is used.

75-77 **ICRNOD (K)**

Number of axial nodes per section of CRD.

78 **NSEGCR**

Number of PRIMAR4 segment representing control rod assemblies.

79 **ICHCHT**

Not currently Used.

80 **NOEQLE**

Controls the pressure in the molten cavity at the time of LEVITATE or PLUTO2 initiation.

= 0, The pressure in the cavity is equal to the DEFORM calculated pressure. If the input **PCFAIL** > 1.0, the pressure is set equal to **PCFAIL**.
= 1, The pressure in the cavity is calculated using the available fission gas.

If LEVITATE or PLUTO2 are initiated after PINACLE, the pressure is always equal to the PINACLE pressure and **NOEQLE** is not relevant.

81 **MTCB**

Core barrel material.

= 1, 316 SS.
= 2, HT-9.

82 **KDEBUG**

= 0, No EBR-II routine debug.
= 1, Turns on EBR-II routine debug.

83 **KEBRS1**

Main time step number to turn EBR-II routine debugging on.

84 **KEBRS2**

Main time step number to turn EBR-II routine debugging off.

85 **IDBDKH**

Reserved for future decay heat modeling.

86 **NULLD3**

Number of flux shape calculations in addition to the initial cold solution in the steady state power distribution/thermal hydraulics iteration for **NPK** > 0.

87 **ISSNUL**

Number of time steps in the steady-state null transient for core channel thermal hydraulics.

If **ISSNUL**=0, no null transient.

Note: **ISSNUL**>0 is required if the multiple pin option is used (**JJMLTP**>0).

88 **IPRSNL**

If **ISSNUL**>0, then print out temperatures every IPRSNL time steps of the null transient.

89 **NOFDBK**

Transient reactivity feedback option for NPK=0.

= 0, Use subroutine FEEDBK for transient reactivity feedbacks.
> 0, Do not calculate transient reactivity feedback with subroutine FEEDBK.

90 **IBOPLT**

Balance-of-plant plotting file option.

= 0, Do not call subroutine LBPLOT.
> 0, Main time step frequency for writing BOP plotting file from subroutine LBPLOT.

91 **ISIMPG**

Graphical interface for simulator application.

= 0, No graphical interface.
= 1, Graphical interface for EBR-II. Some card-image input will be overridden interactively by values from graphical interface. SASSYS-1 is invoked by the graphical interface, not directly by the user. Usage instructions are in ANL-FRA-171.

92 **KHDBK**

Print flag for messages if any extrapolation of the IFR handbook data is used. Only for **IMETAL**=2, **IRHOK**=1 and **IFUELM**=0.

= 0, Print messages.
= 1, Do not print messages.

93 KFIRR

Correction option for irradiation effect on metal fuel thermal conductivity.

= 0, Empirical correction based on average fuel burnup (See **BURNFU**).
= 1, Theoretical correction based on porosity and sodium logging of fuel radial node. (See **PRSTY** and **XLOGNA**).

94 KDENBU

Option for correction of U-Pu-Zr alloy fuel theoretical density for the presence of fission products, based on average fuel burnup and fabricated composition (See **BURNFU**, **WUREF**, **WPUREF**, **WZRREF**). Only for **IMETAL** = 2, **IRHOK** = 1, and **IFUELM** = 0.

= 0, No correction for fission products.
= 1, Make correction for fission products.

95-100 IFIT (K)

Input table look-up options.

IFIT:

= 0, Linear fit.
= 1, Third order fit.
= 2, Third order fit with slope discontinuities if: X(J+1) - X(J) < 0.001.
= 3, Linear fit to log(Y).
= 4, Third order fit to log(Y).
= 5, Third order fit to log(Y) with slope discontinuities.

K:

= 1, Power vs. time or user specified reactivity vs. time curves (**PREATB**, **PRETB2**).
= 2, PRIMAR-1 pump head vs. time or channel flow vs. time.
= 3, PRIMAR-1 inlet temperature vs. time.
= 4, PRIMAR-4 pump head, motor torque, or pump speed vs. time.
= 5, Reserved for future use.
= 6, Reserved for future use.

101 **IPIC**

PINACLE/LEVITATE pin picture plot flag.

= 0, No plot data on unit 19.
= 1, Plot data written on unit 19.

102 **IBOWTP**

Temperature flag in EBR-II bowing reactivity.

= 0, Normalized core temperature rise based on upper plenum temperature.
= 1, Normalized core temperature rise based on average coolant temperature in fueled channels.

103 **ITARGE**

Not currently used.

104 **IEMGEM**

FFTF Gas Expansion Module (GEM) model.

= 0, Not used.
> 0, Invoke FFTF GEM model.

105 **NEXSO**

Number of entries in **EXSOTB**/**EXSOTM** point kinetics external source table (Max. 20).

106 **IPBDEN**

Heavy liquid metal coolant thermophysical properties. (Overridden by **KPROPI** and **ICLPRP**).

= 0, Coolant properties determined by **INAS3D** and **ID2O**.
= 1, Use Pb coolant thermophysical properties.
= 2, Use Pb (44.5%)-Bi (55.5%) eutectic thermophysical properties.

107 **MSTPLA**

Transient time step for change of plotting data output frequency from **MSTPL1** to **MSTPL2**.

108 MSTPLB

Transient time step for change of plotting data output frequency from **MSTPL2** to **MSTPL3**.

109 MSTPL1

Transient time step plotting data output frequency before time step **MSTPLA**.

110 MSTPL2

Transient time step plotting data output frequency after time step **MSTPLA** and before time step **MSTPLB**.

111 MSTPL3

Transient time step plotting data output frequency after time step **MSTPLB**.

112 KTREAT

TREAT modeling flag.

= 0, Use standard modeling.
> 0, Use special TREAT modeling.(Air properties for coolant, constant coolant flow rates).

113 KQSCRA

Modeling option for external source with scram.

= 0, Trip external source at time TSCRAM+DELSCR.
= 1, Hold external source at constant value after TSCRAM+DELSCR.
= 2, Continue external source according to **NEXSO**, **EXSOTB**, **EXSOTM** specifications after TSCRAM+DELSCR. If NEXSO = 0, the external source is held at the initial steady-state value set by **RHOZRO**.

114 KPROPI

Coolant thermophysical properties correlation coefficients. (Overridden by **ICLPRP**).

= 0, Use default coefficients defined by **INAS3D**, **ID2O**, **IPBDEN**, or **ICLPRP**.
> 0, Input correlation coefficients in **APROPI**.

115 ISCH

Coolant channel thermal-hydraulics model option.

= 0, Use single-pin model.
> 0, Use detailed coolant sub-channel model.

116 ISKDOT

Input data printing option.

= 0, Print data as input, and also print the full input data file as stored in memory (DATOUT).
> 0, Skip the DATOUT print, do not recapitulate the input data.

117 NLINMX

Printed output line limit, in thousands of lines, for the printed output file.

No longer used.

118 ICLPRP

Coolant material properties used in the core and all of PRIMAR except for the DRACS loops.

= 0, Correlation coefficients set by **INAS3D**, **ID2O**, **IPBDEN**, or **KPROPI**
= 1, For Na, SAS4A version.
= 2, For Na, SAS3D version.
= 3, For NaK.
= 4, For D2O.
= 5, For Pb.
= 6, For Pb-Bi.
= 7, For user-supplied property correlation coefficients set by **APROPI**.

Note: If **ICLPRP** > 0, then **ICLPRP** overrides **INAS3D**, **ID2O**, **IPBDEN**, and **KPROPI**. See **INAKDR** for the DRACS loop properties.

119 IFT1TM

Output flag for printing maximum core channel temperature data to auxiliary output file MaxTemps.txt.

= 0, Do not write maximum core channel temperature data.
> 0, Write maximum core channel temperature data every **IFT1TM** time steps.

120 **IVIS3D**

3D visualization data output flag. Normally used for detailed sub-channel output.

= 0, Do not write visualization data.
> 0, Write visualization data every `IVIS3D` time steps.

121-250 **INPDU2**

Not currently used.

2.7.2 Block 3 — INPMR4 — PRIMAR-4 Integer Input

1 **NCVP**

Number of compressible volumes, primary loops.

2 **NCVS**

Number of compressible volumes, secondary loops.

3 **NCVD**

Number of compressible volumes, DRACS loops.

Not implemented. See instead `ISGLNK` and `ICVNAK`.

4 **NSEGLP**

Number of liquid segments, primary loops. All core channels (except the bypass channels) form one liquid segment.

5 **NSEGLS**

Number of liquid segments, secondary loops.

6 **NSEGLD**

Number of liquid segments, DRACS loops.

Not implemented. See instead `ISGLNK` and `ICVNAK`.

7 **NSEGGP**

Number of gas segments, primary loops.

8 NSEGGS

Number of gas segments, secondary loops.

9 NSEGGD

Number of gas segments, DRACS loops.

Not implemented. See instead **ISGLNK** and **ICVNAK**.

10 NELEMT

Total number of liquid flow elements, max = 140. A bypass channel must not be split into more than one flow element.

Maximum total numbers:

Item	Limit
Compressible volumes	38
Liquid segments	40
Gas segments	28
Liquid elements	140
Pumps, sodium	12
HXs	4
Steam generators	
Table look-up model	12
Detailed model	8
Check valves	6
DRACS heat exchangers. Not implemented. See instead **ISGLNK** and **ICVNAK**.	4
Temperature groups	100
Bypass channels	8
Axial nodes in HX	61

Array subscripts used in this block:

Subscript	Description
ICV	Compressible volume
ISGL	Liquid segment
ISGG	Gas segment
IELL	Liquid flow element
M	1 Inlet 2 Outlet
IPMP	Pump
IIHX	HX
ITGP	Temperature group
IBYP	Bypass channel
ISGN	Steam generator
IVLV	Valve
ICKV	Check valve
IDHX	Air dump heat exchanger
IRVC	RVACS

11-48 ITYPCV (ICV)

Compressible volume type.

=1, Inlet plenum.
=2, Compressible liquid volume, no gas.
=3, Compressible outlet plenum, no cover gas.

=4, Almost incompressible liquid, no gas.
(pipe tees, extra inlet plenum, etc.)
=5, Pipe rupture source.
=6, Pipe rupture sink, guard vessel.
=7, Outlet plenum with cover gas.
=8, Pool with cover gas.
=9, Pump bowl and cover gas.
=10, Pressurizer with cover gas.

=11, Compressible gas volume, no liquid.

49-188 ITYPEL (IELL)

Liquid flow element type.

=1, Core subassemblies.
=2, Bypass channel.
=3, Pipe.

=4, Check valve.
=5, Pump.
=6, HX, shell side.
=7, HX, tube side.
=8, Steam generator, sodium side.
=9, DRACS heat exchanger, tube side. (Not implemented. Use **ITYPEL** = 7. See **ISGLNK**.)
=10, DRACS heat exchanger, shell side. (Not implemented. Use **ITYPEL** = 6. See **ISGLNK**.)
=11, Valve.
=12, Air dump heat exchanger, sodium side.
=13, Annular element.
=15, Annular pump.

189-268 **JCVL (M,ISGL)**

Compressible volumes at the ends of the liquid segment.

269-324 **JCVG (M,ISGG)**

Compressible volumes at the ends of the gas segment.

325-364 **NELML (ISGL)**

Number of elements in the liquid segment.

365-404 **JFSELL(ISGL)**

First element number in segment ISGL. Segment ISGL contains elements **JFSELL**(ISGL) through **JFSELL**(ISGL)+**NELML**(ISGL)-1.

PUMPS

405 **NPUMP**

Number of sodium pumps: Max 12.

406-417 **IELPMP (IPMP)**

Element number of pump IPMP.

418-429 **IEMPMP (IPMP)**

Type of pump.

=-3, *Equivalent Circuit EM Pump*.
=-2, *ALMR EM pump*.
=-1, *Electromagnetic pump*.
=0, *Use table of pump head vs time*.

=1, *Centrifugal pump*.
=2, *Homologous pump model*.
=3, *EBR-II pump model*.

430-469 **ILRPMP (IPMP)**

= 0, Pump operation according to model selected, see **IEMPMP**. Uses locked rotor model for **IEMPMP** = 2 according to pump WB and SB, see **APMPHD**. Uses static trip time in **APMPHD** (7, IPMP) when **IEMPMP** = -2.
= 1, Pump speed set to zero, locks rotor immediately as in a pump seizure. Applicable when **IEMPMP** = 1 or 2.
=-1, For table of pump speed vs. time (**IEMPMP** = 1 or 2).
=-2, For table of pump *head vs. flow*. This option overrides input for **IEMPMP**.
>0,

IEMPMP	Description
-2	ID for function that returns the trip time for the *ALMR EM Pump model*. A single argument is passed containing the value of the time at the end of the current PRIMAR-4 time step. When a function is used, **APMPHD** (7,IPMP) is ignored. (See *FUNCTION Block*)
-3	*EMPUMP Block* number containing pump description. Values in **PMPINR**(IPMP), **HEADR**(IPMP), **PMPSPR**(IPMP), **PMPFLR**(IPMP), **PMPEFR**(IPMP), **TRKLSC**(IPMP), **EPSCAV**(IPMP), **APMPHD**(K,IPMP), and **AMOTTK**(K,IPMP) will be ignored.

HX, DRACS, CHECK VALVES

470 **NIHX**

Number of IHXs and PHXs: Max. 4.

471 **NDRACS**

Number of DRACS heat exchangers.

Not implemented. See instead **ISGLNK** and **ICVNAK** and include with **NIHX**.

472 **NCKV**

Number of check valves: Max. 6.

473-476 **IELIHX (IIHX)**

Element number of HX # IIHX, primary loop.

477-480 **IELDRP (IDRX)**

Element number of DRACS # IDRX, primary loop.

Not implemented. See instead **ISGLNK** and **ICVNAK** and use **IELIHX**.

481-484 **ILIHXS (IIHX)**

Index of the secondary element in the IIHX heat exchanger.

The sign of the index indicates what detailed heat exchanger model is used.

> 0, Detailed Intermediate Heat Exchanger
< 0, Detailed Primary Heat Exchanger

485-488 **IELDRS (IDRX)**

Element number of DRACS # IDRX, intermediate loop.

Not implemented. See instead **ISGLNK** and **ICVNAK** and use **ILIHXS**.

489-492 **IHXCLC (IIHX)**

= 0, Use detailed HX. **NTNODE** must be greater than or equal to 2.
≠ 0, Use table look-up IHX. **NTNODE** must equal 2.

> 0, Table no. ITAB for relative temperature drop table look-up IHX.
< 0, Table no. -ITAB for exit temperature table look-up IHX.

If ITAB <= 12, see **DTMPTB**, **ZCENTR**, **TMPMTB**. ITAB specifies both the **DTMPTB** and the **ZCENTR** table number unless **iHXZID** is provided.
If ITAB > 12, *Function ID* ITAB defines the temperature boundary condition. **iHXZID** must be provided to define the height of thermal center.

493-496 **IDRCLC (IDRX)**

= 0, Use detailed DRACS.
> 0, Table no. ITAB for table look-up DRACS. (See **DTMPTB**, **ZCENTR**, and **TMPMTB**).

Not implemented. See instead **ISGLNK** and **ICVNAK** and use **IHXCLC**.

497 **IPRADJ**

= 0, No PRMADJ pressure adjustments for channel flow estimation errors.
=1, Adjustments for the outlet plenum only. Recommended value.
=2, Adjustments for both inlet and outlet plenums.

DEBUG PRINTS

498 **ICPDBG**

= 0, No pump debugs.
= 1, Pump debugs.
> 1, Extended pump debugs for centrifugal pumps and EQ EM pumps.

498-506 **IBL3D2 (I)**

Inlet and outlet pressure debugs if **IBL3D2**(1) > 0. **IBL3D2**(2) > 0 for steady- state core outlet temperature debugs.

507 **IDBPR4**

PRIMAR4 debug parameter, initial value.

=0, No debug print-out.
=1, Final results for each PRIMAR step.

508 **IDBP4N**

Value of **IDBPR4** after time = **TMDBP4**.

509 **INAKDR**

Coolant properties for the DRACS loops. DRACS liquid segments and compressible volumes are defined by **ISGLNK** and **ICVNAK**, respectively.

=0, Use the same coolant properties in all loops, as determined by **INAS3D**, **ID2O**, **IPBDEN**, **KPROPI**, or **ICLPRP**.
=1, Use SAS4A correlations for Na in the DRACS loop.
=2, Use SAS3D correlations for Na in the DRACS loop.
=3, Use NaK properties in the DRACS loop.
=4, Use D2O properties in the DRACS loop.
=5, Use Pb properties in the DRACS loop.
=6, Use Pb-Bi properties in the DRACS loop.
=7, User supplied property correlation coefficients.

PIPE RUPTURE

510 **ISRCRP**

Compressible volume number of pipe rupture source.

511 **ISNKRP**

Compressible volume number of pipe rupture sink.

TEMPERATURE GROUPS

Elements in a liquid segment are combined into temperature groups. Each temperature group contains one or more consecutive elements. All of the elements in a temperature group are treated with the same type of liquid temperature calculation: as a pipe, HX, bypass channel, or steam generator. See *Liquid Temperature Calculations*.

512 **NTGPT**

Number of temperature groups. **NTGPT** ≤ 100.

- Core channel segments must not be assigned to a temperature group.
- It is recommended to combine consecutive pipe-type elements within a segment as a single temperature group.
- An element representing the primary side of a detailed heat exchanger must be assigned to its own temperature group. The secondary side will inherit the same group and must not be included in another temperature group.
- Each element representing a bypass channel must be assigned to its own temperature group.
- Elements with direct coolant and/or wall *heat* must be assigned to their own temperature groups.

513-612 **NTNODE (ITGP)**

Number of nodes in the temperature group.

NTNODE >= 2.

Table look-up heat exchangers and steam generators must have exactly two nodes.

613-712 **IFSTEL (ITGP)**

First element in the temperature group.

713-812 **ILSTEL (ITGP)**

Last element in the temperature group.

BYPASS CHANNELS

813 **NBYP**

Number of bypass channels.

814-821 **NTLWBY (IBYP)**

Number of nodes in lower part of walls A and B of bypass channel - Region 1.

822-829 **IDKTYP (IBYP)**

Decay heat curves for bypass channels.

830-837 **IELBYP (IBYP)**

Element numbers for bypass channels. (Usually opposite active core).

838 **ISSTP**

Not currently used.

STEAM GENERATORS

839 **NSGN**

Number of steam generators.

840-851 **IELSGN (ISGN)**

Element number for steam generator.

852-863 **ISGCLC (ISGN)**

= 0, Use detailed steam generator.

≠ 0, Use table look-up steam generator. **NTNODE** must equal 2.

> 0, Table no. ITAB for relative temperature drop table look-up steam generator.

< 0, Table no. -ITAB for exit temperature table look-up steam generator.

If ITAB <= 12, see **DTMPTB**, **ZCENTR**, **TMPMTB**. ITAB specifies both the **DTMPTB** and the **ZCENTR** table number unless **iSGZID** is provided.

If ITAB > 12, *Function ID* ITAB defines the temperature boundary condition. **iSGZID** must be provided to define the height of thermal center.

864-875 **IEVAP (ISGN)**

For all steam generator models,

= 1, Evaporator.

= 2, Superheater.

= 3, Once-through.

CHECK VALVES

876-881 **IELLCK (ICKV)**

Element number for check valve ICKV.

882-889 **IUM883**

Not currently used.

PRINTS AND BINARY OUTPUT

890 **IP4PRT**

Print PRIMAR-4 results every **IP4PRT** PRIMAR steps.

891 **NBINOT**

Number of **IBINOT** entries for PRIMAR-4 binary output PRIMAR4.dat. If **NBINOT** = 0, no output.

892 **IBINST**

PRIMAR-4 binary output every **IBINST** steps.

893-971 **IBINOT**

Identification of binary output for PRIMAR4.dat. First two digits give IBNTYP = type of variable. Last 4 digits give INUM, the variable subscript. If INUM > 5000, then INUM-5000 is the starting value for a range of subscripts, and the next INUM is the last value in the range. See Table 2.7.1 and Table 2.7.2 for description of input options.

972 **INULLT**

Steady-state null transient plotting flag. Used with **NBINOT** and **IBINOT**.

=0, No steady-state plot information written to PRIMAR4.dat.

> 0, Write steady state plot information to PRIMAR4.dat. For **ISSCPC** > 0, plot information is written to PRIMAR4.dat every **INULLT** steps during the null transient.

VALVES

973 **NVALVE**

Number of valves, ≤ 8,

974-981 **IELVLV (IVLV)**

Element number for valve IVLV.

982-989 **ITABVV (IVLV)**

Table number in DTMPTB tables for valve pressure drop coefficient vs. time.

Note: Enter the initial pressure drop coefficient for the value **G2PRDR** (**IELVLV**(IVLV)).

PUMP DEFAULTS

990 **IPMDFT**

≤ 0, Default values are used for **PMPHD** and **PMPTQ**.
> 0, User-defined values are used for **PMPHD** and **PMPTQ**.

Only used when **IEMPMP**(IPMP) = 2.

AIR DUMP HEAT EXCHANGERS

991 **NDHX**

Number of air dump heat exchangers. Max. = 4.

992-995 **IELDHX (IDHX)**

Element number for air dump heat exchanger IDHX.

996-999 **IFCDHX (IDHX)**

= 0, Natural convection.
= 1, Forced convection on the air side.

DETAILED STEAM GENERATOR

1000-1003 **IFWC (ISGN)**

Feedwater control options.

= 1, Constant feedwater flow at steady state value.
=2, Table look-up of feedwater flow.
=3, Drum level controller.

1004-1007 **IGHC (ISGN)**

Specification of multiple evaporator and superheater sections.

= Number of superheater parallel sections X 100 + number of evaporator parallel sections.

CONTROL ROD DRIVE EXPANSION REACTIVITY FEEDBACK

1008 **NEXPFB**

= 0, No contribution to control rod expansion feedback from vessel wall heating.
> 0, Number of liquid elements and/or compressible volumes contributing to control rod expansion feedback.

1009-1018 **IEXPFB (K)**

> 0, Element number.

< 0, -Compressible volume number.

PLENUM MODEL

1019 **IPL2A**

Not currently used.

COMPONENT-COMPONENT HEAT TRANSFER

1020-1051 **IBYBY (K,IBYP)**

The K-th bypass channel number for subassembly to subassembly heat transfer from bypass channel IBYP. If -NCH ≤ **IBYBY** < 0, then -**IBYBY** is a core channel number. If **IBYBY** < -NCH, then -**IBYBY** is the temperature of a constant temperature heat sink. Dimension (4,8).

1052-1081 **IELHT (K)**

Element number of the K-th element involved in component-to-component heat transfer from **IELHT**(K) to **IELHT2**(K). For the second wall of annular element IELL, **IELHT** = 1000 + IELL.

1082-1111 **IELHT2 (K)**

Element number.

ICV, or

Temperature of heat sink (if -IELTH2 > max number of CVs.).

1112-1141 **NELHTN (K)**

+ N, Use first N nodes.

- N, Use last N nodes.

=0, Use all nodes.

1142 **IDBHTH**

= 0, No debug prints.

=1, Short comp-comp prints.

≥ 2, Debug prints.

1143 **ISTHTH**

PRIMAR step when IDVHTH is turned on.

1144-1152 **IB3DM2**

Not currently used.

COMMON COVER GAS PRESSURES

1153 **NCCV**

Number of connected compressible volume cover gasses with common gas pressure.

1154 **ICCVFS**

First compressible volume with common gas pressure.

STEADY-STATE INITIALIZATION

1155-1158 **ISSIHX (IIHX)**

Steady state temperature drop.

= 1, If user specifies.

1159-1170 **ISSPMP (IPMP)**

Steady state pump head.

= 1, If user specifies.

1171 **NIHXBY**

Number of liquid segments that bypass the IHX.

1172-1181 **IHXBYP (K)**

Liquid segment numbers for the segments that bypass the IHX.

1182 **NPMPBY**

Number of liquid segments that bypass the pump.

1183-1192 **IPMPBY (K)**

Liquid segment numbers for the segments that bypass the pump.

BYPASS CHANNEL HEAT TRANSFER

1193-1200 **IHTBYB (IBYP)**

Coolant heat transfer parameter set for wall B in bypass channel IBYP.

1201-1208 **IHTBYD (IBYP)**

Coolant heat transfer parameter set for wall D in bypass channel IBYP. See **C1BY**, **C1BY2**, **C1BY3**, and **C1BY4**.

FORMER CONTROL ROD DRIVE FEEDBACK

1209-1242 **KCHUIS**

Reserved. (was **ICHUIS** in Version 2.1).

ANNULAR ELEMENTS

1243 **NANEL**

Number of annular elements.

1244-1273 **IELANE (IANL)**

Element number for annular element.

RVACS INPUT

1274 **NSCRVC**

Number of sections in the RVACS < 7.

1275 **IRVOPT**

RVACS modeling option:

=0, Use the detailed RVACS model.

=1-12, Use the simple RVACS model. IRVOPT is the number of entries in the table of h vs. T (see **RVHTAB** and **RVHTMP**).

>12, Use the simple RVACS model. IRVOPT is the *FUNCTION* ID for h(T). See also **IRVHTS**.

=-1, Use the coupled RVACS model. External code participates in the null transient.

=-1000, Use the coupled RVACS model. External code does not participate in the null transient.

= -1001 - -1012, Use the coupled RVACS model during transient, and the simple RVACS model during the null transient. ABS(IRVOPT+1000) is the number of entries in the table of h vs T.

< -1012, Use the coupled RVACS model during the transient, and the simple RVACS model during the null transient. ABS(IRVOPT+1000) is the *FUNCTION* ID for h(T) during the null transient.

1276-1281 **IELRVC (IRVC)**

Element number or -ICV, starting at the bottom and going up. If **IELRVC**(IRVC) > 1000, use second wall of element IELVRC(IRVC)-1000.

1282-1287 **NANRVC (IRVC)**

Number of nodes in this section, only applicable if CV.

NULL TRANSIENT

1288 **ISSCPC**

Number of time steps in the null transient to initialize component-component heat transfer and *direct heating*.

1289 **ISST15**

Print PRIMAR-4 results every `ISST15` steps during the null transient.

RVACS INPUT

1290 **IDBRV**

Debug parameter for RVACS.

=0, No debug.
=1, Regular print every time step.
=2, Detailed debug prints.

1291 **IDBRVS**

PRIMAR-4 step when RVACS debugs start.

RADIAL REFLECTOR REACTIVITY FEEDBACK

1292 **NRREAC**

Number of radial reflectors involved in reactivity calculations. (for use with empirical feedback models).

1293-1300 **ISLREA (K)**

Segment numbers of radial reflectors for reactivity calculations.

1301 **LBYP**

Number of radial reflector bypass channels.

1302-1309 **LELBYP**

Element number of the radial reflector bypass channels.

PIPE TEMPERATURE OPTION

1310 IPIPTM

PRIMAR-4 pipe temperature convective term differencing approximation.

=0, Always use a Lagrangian calculation for coolant temperatures in pipes and annular elements.
=1, Use an Eulerian calculation for large time steps (the coolant moves two or more nodes in a time step).
=2, Use an Eulerian calculation for large time steps only during the PRIMAR-4 null transient (**ISSCPC**>0).

Notes:

1) The Eulerian calculation is faster for large time steps.
2) The Lagrangian calculation does not result in numerical axial diffusion, whereas the Eulerian calculation does.
3) The Eulerian calculation is not compatible with *direct heat* modeling.
4) Recommended value: **IPIPTM**=2.

MULTIPLE INLET/OUTLET PLENA

1311 IFMIOP

Multiple inlet/outlet plenum option.

=0, Single inlet and outlet plena.
=1, Multiple inlet and outlet plena. Must specify **NSEGMP**, **TPLCV**, **PPLCV**, and **ZPLENC**.

ELEMENT/WALL THERMAL ADJUSTMENT

1312 ITHPEN

Optional adjustments to element and wall thermal treatment, based on thermal penetration depth.

=0, No adjustments.
=1, **WALLH** and **HWALL** = thermal conductivity, k.
=2, **WALLH** and **HWALL** = k/total thickness.
=3, **WALLH** and **HWALL** = 3*k/total thickness.

Recalculates **WALLH**, **WALLH2**, **HWALL**, **WALLMC**, **CMWALL**, **WALMC2**, and **HAELHT** for specified elements and compressible volumes. No adjustments are made if **WALTHK**(IELL) = 0, **WALTH2**(IAEL) = 0, or **THKWAL**(ICV) = 0.

STRATIFIED VOLUME MODEL

1313 **NSTRCV**

Number of stratified compressible volumes.

1314-1316 **ICVSTR (ICVST)**

Compressible volume number for stratified treatment.

1317-1319 **ISTRVT (ICVST)**

= 1, For vertical coolant inlet, as in an outlet plenum,
= 2, For horizontal coolant inlet.

1320-1322 **NUMWAL (ICVST)**

Number of wall sections.

1323-1325 **IFSTWL (ICVST)**

Wall number (IW) of the first wall section.

1326-1334 **IWLHRZ (IW)**

= 0, For a vertical wall.
= 1, For a horizontal wall at the top of a compressible volume.
=2, For a horizontal wall at the bottom of a compressible volume.

1335-1343 **NVNDWL (IW)**

Number of vertical nodes in a vertical wall.

NVNDWL = 1 for a horizontal wall.

1344-1352 **NLNDWL (IW)**

Number of lateral nodes in a wall section. Max. = 8. Note: Sum (**NVNDWL***NLNDWL) ≤ 300.

1353-1361 **ICV2WL (IW)**

Number of the compressible volume in contact with the outer side of the wall section. Equal to 0 for an adiabatic outer boundary. If **ICV2WL** > 38, **ICV2WL** = the temperature of a constant temperature heat sink.

1362 **IDBSTR**

Debug flag for the stratified temperature model.

=0, No debug prints.
=1, Final results only.
=5, Everything.

1363 **ISTDBS**

PRIMAR time step when stratified debug starts.

1364 **ISTSTP**

Stop the run at PRIMAR step **ISTSTP**. Not used if **ISTSTP** = 0 or NSTRCV = 0.

1365 **IFT16**

Write out stratified CV output to STRATCV.dat every | **IFT16**| PRIMAR steps. No output if **IFT16** = 0.

>0, No additional debug information printed.
<0, Additional debug information printed.

THICK WALL PIPES

1366 **NTHKPW**

Number of pipes to be treated with a thick wall treatment. Note: Thick wall pipe option is not compatible with *direct element heating*.

1367-1376 **IELTPW (ITWP)**

Element number for thick wall treatment.

DRACS

1377 **ISGLNK**

Use **INAKDR** to determine the coolant properties for ISGL ≥ **ISGLNK** (ISGL = liquid segment number).

Liquid segments in this range are considered to be part of the DRACS.

1378 **ICVNAK**

Use **INAKDR** to determine the coolant properties for ICV ≥ **ICVNAK** (ICV = compressible volume number).

Compressible volumes in this range are considered to be part of the DRACS.

STEADY STATE INITIALIZATION OPTION

1379 **NCVSSI**

Number of compressible volumes for which the steady-state coolant pressure and temperature are specified (Max. = 10).

1380-1389 **ICVSSI (II)**

Compressible volume number for which steady-state pressure and temperature are specified.

AIR DUMP HEAT EXCHANGER OPTION

1390-1393 **IADHX (IDHX)**

= 0, Simple air dump heat exchanger model.
= 1, Code calculates **HOTB**, **HITB**, **XKHXLS**, **XKRLS** as a function of geometry and flow rate.

Not currently used.

1394-1397 **ISTGTB (IDHX)**

= 0, Staggered spacing between tube rows.
= 1, Inline tube bundle.

Used only if **IADHX**(IDHX) = 1.

Not currently used.

1398-1401 **NROW (IDHX)**

Number of tube rows. Used only if **IADHX**(IDHX) = 1.

Not currently used.

1402-1405 **ISSADX (IDHX)**

Steady-state initialization option.

= 0, No steady-state initialization.
=1, Use outlet temperature of element associated with IDHX for initialization.
> 1, Use inlet temperature of element associated with IDHX for initialization.

When ISSADX(IDHX) > 0, the steady-state initialization assumes zero heat removal from the air-dump heat exchanger.

External CFD Coupling for Compressible Volumes

Input for coupling between PRIMAR-4 Compressible Volumes and an external code can be changed during a restart. However, it is up to the user to ensure that the models in force at the end of one simulation and the beginning of the subsequent simulation are consistent. This restart capability is primarily intended to allow a stand-alone, steady-state transient to converge efficiently prior to restarting the simulation with a more computationally intensive CFD model. In this scenario, the input below would not be present in the stand-alone model, but would be present in the restart input.

1406 **NCFDCV**

Number of Compressible Volumes that will be represented by an external CFD model.

$0 \leq$ **NCFDCV** ≤ 4

1407-1410 ICFDCV (I)

Compressible Volume for which an external CFD model will be provided.

0 < **ICFDCV**(I) ≤ NCV

1411 NULLCFD

CFD coupling treatment during Null Transients.

= 0, No coupling during null transients.
> 0, Use scaled-time coupling.

When scaled-time coupling is used, long null transients in SAS4A/SASSYS-1 will be projected to the external CFD models as shorter transients to help minimize CFD overhead during the null transient. Specifically, the CFD time steps will be smaller than or equal to the null transient time steps in SAS4A/SASSYS-1 according to the relation

$$\Delta t_{\text{CFD}} = \frac{\Delta t_{\text{SAS}}}{\text{NULLCFD}}$$

1412 ICFDDBG

Debug flag for CV to CFD coupling.

The complete history of data transferred from SAS4A/SASSYS-1 to an external CFD model is always saved to a separate file. This flag adds additional debugging information to the normal output file.

End of External CFD Coupling for Compressible Volumes

1414 ThickWallTableID

= 0, No thick-walled CV treatment.

> 0, *Table ID* for the thick-walled CV input. A description of the thick-walled input can be found in Table 5.4.10.

1415-1422 iPHXPRP

Secondary coolant type for the Kth HX. Only applicable to a primary heat exchanger.

For available options see **ICLPRP**. Defualt value will be **ICLPRP**.

1423-1430 iPHXTID

Function ID for the inlet temperature boundary condition of the secondary element in the Kth HX. Only applicable to a primary heat exchanger.

1431-1438 **iPHXWID**

Function ID for the mass flow rate boundary condition of the secondary element in the Kth HX. Only applicable to a primary heat exchanger.

1439 **IDRVACSTin**

When using the simple RVACS model:

= 0, Use constant RVACS heat-sink temperature specified by **TAIRVC**.
> 0, *FUNCTION* ID for RVACS heat-sink temperature as a function of time.

When using the detailed RVACS model:

= 0, Use constant RVACS air inlet temperature specified by **TAIRVC**.
> 0, *FUNCTION* ID for RVACS air inlet temperature as a function of time.

If IDRVACSTin > 0, **TAIRVC** will be ignored by the code.

1440 **IDRVACSKin**

= 0, Use constant RVACS air inlet orifice coefficient specified by **ORFIN**.
> 0, *FUNCTION* ID for RVACS air inlet orifice coefficient as a function of time.

If IDRVACSKin > 0, **ORFIN** will be ignored by the code.

1441 **SegLossCoefTableID**

> 0, *Table ID* referencing segment inlet anisotropic Re-dependent loss coefficient data.

A sample input table is shown in Listing 5.3.2.

1442 **ElILossCoefTableID**

> 0, *Table ID* referencing element outlet anisotropic Re-dependent loss coefficient data.

A sample input table is shown in Listing 5.3.1.

1443-1446 **iHXZID**

Table no. ITAB defining height of thermal center for table look-up IHX (see **IHXCLC**).

<= 12, see **ZCENTR** and **TMPMTB**.
> 12, *Function ID* ITAB defines the height of thermal center.

1447-1462 **NONE**

Not currently used.

1451-1462 **iSGZID**

Table no. ITAB defining height of thermal center for table look-up steam generator (see **ISGCLC**).

<= 12, see **ZCENTR** and **TMPMTB**.
> 12, *Function ID* ITAB defines the height of thermal center.

1463-1474 **NONE**

Not currently used.

1475 **CoolHeatTableID**

= 0, No direct coolant heating.

> 0, *Table ID* for direct coolant heating input.

An example table is shown in Listing 5.4.1

1476 **WallHeatTableID**

= 0, No direct wall heating.

> 0, *Table ID* for direct wall heating input.

An example table is shown in Listing 5.4.2

1477 **ILGVHT**

Option to specify whether the first element in the RVACS model will be treated as a lower cavity mixing region. See Figure 5.4.9.

= 0, Do not include the lower cavity mixing region (default).
= 1, Include the lower cavity mixing region.

See also **SPLITF** and **HTCLGV**.

1478 **IRVHTS**

FUNCTION ID for the simple RVACS heat transfer coefficient scaling factor as a function of time.

If **IRVHTS** = 0, the scaling factor defaults to one. See Eq. (5.4-106).

See also **RVHTAB** and **RVHTMP** or **IRVOPT** > 12.

1479-1600 **IBL3DM**

Not currently used.

Table 2.7.1: Input in INPMR4 to Control Binary Output in PRIMAR.dat

Variable	Location	Description
NBINOT	891	The number of IBINOT entries for the binary output file. If NBINOT=0, no PRIMAR4.dat output will be generated.
IBINST	892	Output every IBINST PRIMAR time steps, default=1
IBINOT(K), K= 1:NBINOT	893-972	Identification of binary output. IBINOT is a 6-digit code where • The first 2 digits give IBNTYP, the type of variable (or data item) • The last 4 digits give the subscript INUM for the element, segment, temperature group, etc. There is a special case where INUM>5000, in which case INUM-5000 is the starting value for a range of subscripts and the next INUM is the last value in that range. *Examples*: 240006: • IBNTYP=24, INUM=6. Print the liquid temperature of CV_6. 10015: • IBNTYP=1, INUM=15: Print the liquid segment flow of S_{15} 245001 08: • First IBNTYP=24, INUM=5001 • Second IBNTYP is blank, INUM=8 • Print the liquid temperature of a range of compressible volumes from CV_1 to CV_8.

The notation for the subscripts in Table 2.7.2 is as follows:

- ISGL = liquid segment
- ISGG = gas segment
- IELL = liquid element
- ICH = channel
- ICV = compressible volume
- IDHX = air dump heat exchanger
- INOD = node
- ITGP = temperature group
- IPMP = pump
- K, J = context specific
- L = 1 for inlet, 2 for outlet

Table 2.7.2: IBNTYP and INUM Codes to Request Edits on PRIMAR4.dat

IBN-TYP	INUM	Description	Variable
1	ISGL	flow, liquid segment	FLOSL2(ISGL)
2	ISGG	flow, gas segment	FLOSG4(ISGLG)
3	ICH	estimated channel inlet	CHFLO2(1,ICH)
4	ICH	estimated channel outlet flow	CHFLO2(2,ICH)
5	L	estimated core flow	CORFLE(L)
6	L	estimated core flow times temperature	CORFTE(L)
7	L	actual integrated channel flow	CORCHF(L)
8	L	channel flow times temperature	CORFLT(L)
9	ICH+100*(L-1)	actual channel flow, beginning of step	FLOCH1(L,ICH)
10	ICH+100*(L-1)	coefficients used to estimate the	C0FLCH(L,ICH)
11	ICH+100*(L-1)	core flow for the next step	C1FLCH(L,ICH)
12	ICH+100*(L-1)		C2FLCH(L,ICH)
13	ICH+100*(L-1)		C3FLCH(L,ICH)
14	ICH+100*(L-1)	subassembly inlet or outlet temperature	TEXPEL(L,ICH)
15	ICH+100*(L-1)	energy of vapor condensing in inlet or outlet plenum	ENVAPR(L,ICH)
16	ICV	liquid pressure for compressible volume ICV	PRESL2(ICV)
17	ICV	gas pressure	PRESG2(ICV)
18	IPMP	pump head for pump IPMP	HEADP2(IPMP)
19	ICV	cover gas interface height	ZINTR2(ICV)
20	ICV	gas volume	VOLGC2(ICV)
21	ICV	total volume, liquid+gas	VOLLGC(ICV)
22	ICV	liquid mass	XLQMS2(ICV)
23	ICV	gas mass	GASMS2(ICV)
24	ICV	liquid temperature	TLQCV2(ICV)
25	ICV	liquid density	DNSCV2(ICV)
26	ICV	wall temperature	TWLCV2(ICV)
27	ICV	gas temperature	TGASC2(ICV)
28	ISGL+100*(L-1)	liquid segment inlet or outlet temperature	TSLIN2(L,ISGL)
29	IELL	gravity head for element IELL	GRAVHD(IELL)
30	IELL+400*(L-1)	liquid element temperature	TELEM(L,IELL)
31	ITGP	fraction of a node traversed by Lagrangian slugs In temperature group ITGP	FRNDF2(ITGP)
32	INOD	liquid temperature, node INOD	TLNOD2(INOD)
33	INOD	wall temperature	TWNOD2(INOD)
34	–	outlet plenum density	DLHOT
35	–	inlet plenum density	DLCOLD
36	ICH	inlet temperature	TINVAL(ICH)
37	–	outlet plenum pressure at beginning of time step	PXT0
38	–	time derivative of outlet plenum pressure	DPXDT
39	IPMP	pump speed	PSPED2(IPMP)
40	–	time derivative of inlet plenum pressure	DPINDT
41	–	inlet plenum pressure at beginning of time step	PIN
42		Not used	
43		Not used	
44		Estimated boiling time	BOILTM
45	–	next PRIMAR step size	DTPNXT

continues on next page

Table 2.7.2 – continued from previous page

IBN-TYP	INUM	Description	Variable
46	ICH	coolant re-entry temperature	TUPLVL(ICH)
47		Not used	
48	L	accumulated error in plenum mass	DMSSUM
49	L	accumulated error in plenum mass times temperature	DMTSUM
50	INOD	temperature of sink for component-to-component heat transfer	TSNKND(INOD)
51	INOD	heat transfer coefficient for component-to-component heat transfer	HSNKND(INOD)
52	ICV	temperature of sink for component-to-component heat transfer	TSNKCV(ICV)
53	ICV	heat transfer coefficient for component-to-component heat transfer	HSNKCV(ICV)
54	ICV	component-to-component heat transfer rate from compressible volume ICV	QSNKCV(ICV)
55	–	RVACS heat removal rate	QRVACS
56	K	component-to-component heat transfer rate for path K	QCPCP(K)
57	–	RVACS air flow rate	WAIRV2
58	K	RVACS temperature for node K	TRVACS(K)
59	K	guard vessel temperature	TW2RV2(K)
60	K	shell inner temperature	TW3RV2(K)
61	K	shell outer temperature	TW4RV2(K)
62	K	outer wall temperature	TW5RV2(K)
63	K	temperature of air between guard vessel and shell	TA1RV2(K)
64	K	temperature of air between shell and outer wall	TA2RV2(K)
65	IPMP	pump torque for pump IPMP	TQMB3(IPMP)
66	–	Net reactivity	REANET
67	–	reactivity from external function (PREA)	REAPRO
68	–	reactivity from CRDL expansion and scram reactivity	REASCR
69	–	Doppler reactivity feedback	READOP
70	–	Fuel axial expansion reactivity feedback	READEN
71	–	Radial expansion reactivity feedback	REAREX
72	–	Coolant voiding reactivity feedback	REACOL
73	–	Fuel relocation reactivity feedback	REAFUL
74	–	Clad relocation reactivity feedback	REACLD

continues on next page

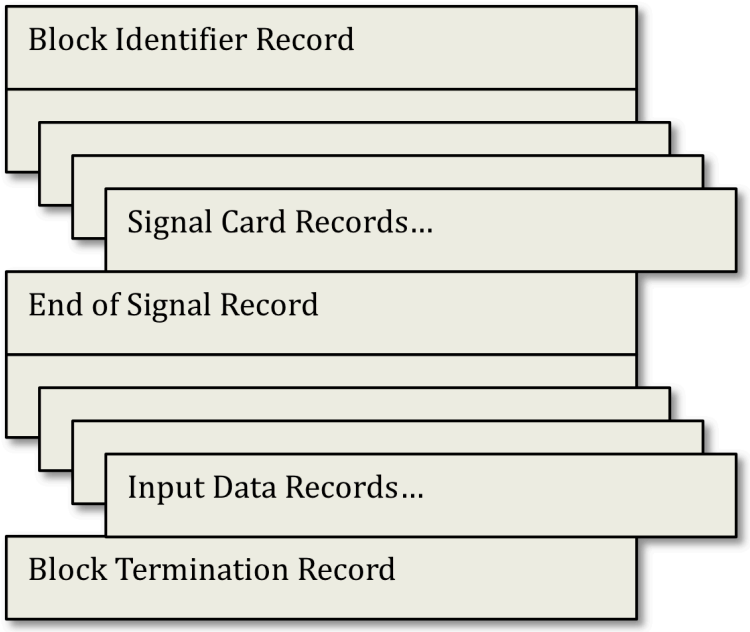
Table 2.7.2 – continued from previous page

IBN-TYP	INUM	Description	Variable
75	J	Advanced user option for channel dependent reactivity feedback • If J < 826, coolant voiding reactivity feedback for channel J • If 825 < J < 1651, fuel relocation reactivity feedback for channel J - 825 • If 1650 < J < 2476, clad relocation reactivity feedback for channel J - 1650 • If 2475 < J < 3301, total axial expansion reactivity feedback for channel J - 2475 • If 3300 < J < 4126, Doppler reactivity feedback for channel J - 3300	REAICH(J)
76	J	Normalized decay power for region J	POWDKH(J)
77	IDHX	Heat removal rate for ADHX IDHX	QQ(IDHX)
78	IDHX	Air flow rate for ADHX IDHX	WAIR(IDHX)
79	IDHX	Air outlet temperature for ADHX IDHX	TAIROT(IDHX)
80	IELL	Pressure drop in element IELL, excludes gravity head	DPRSEL(IELL)
81	IPMP	Voltage in EM pump IPMP	EMVOLT(IPMP)
82	IPMP	Frequency in EM pump IPMP	EMFREQ(IPMP)
83	IPMP	Slip in EM pump IPMP	EMSLIP(IPMP)
84	IPMP	Current in EM pump IPMP	EMCURR(IPMP)
85	IPMP	Phase in EM pump IPMP	EMPHAS(IPMP)
86	IPMP	Coolant Heat Generation Rate in EM pump IPMP	EMPCHR(IPMP)
87	IPMP	Wall Heat Generation Rate in EM pump IPMP	EMPWHR(IPMP)
88	iEll	Direct coolant heat in element iEll.	CLHEAT(iEll)
89	iEll	Direct wall heat in element iEll. If iEll is greater than 1000, the second wall of element iEll-1000 is identified.	WLHEAT(iEll)

2.7.3 Block 5 — INCONT — Control System Input

INPUT BLOCK STRUCTURE

The control system input block structure is identical to the standard SAS4A/SASSYS-1 input block structure in all but one respect. A new card format known as a signal card has been introduced. These cards immediately follow the block identifier card and precede the standard format data cards. The ordering of the different card types is depicted as follows:



The format for the block identifier card is described *above*. For the control system input block the block name is "INCONT" and the block number is 5.

SIGNAL CARDS

A signal card contains data fields for the FORTRAN variables ISIG, JTYPE, J1SIG, J2SIG, F1SIG, F2SIG, F3SIG, F4SIG, and F5SIG, with record format 4I5,5F10.3. These variables are described in Table 2.7.3.

A signal card is used to define a signal in the user's block diagram. There are four signal types: measured, demand, block, and control. Each signal must be assigned a unique signal identification number using the "ISIG" field. The value of ISIG must be greater than zero and less than 999, or between 1000 and 10000.

Table 2.7.3: Signal Card Format

Column	Fortran Symbol	Definition	Variable Type
1-5	ISIG	Serial Number	Integer
6-10	JTYPE	Signal Type	Integer
11-15	J1SIG	Signal Descriptor 1	Integer
16-20	J2SIG	Signal Descriptor 2	Integer
21-30	F1SIG	Constant 1	Real
31-40	F2SIG	Constant 2	Real
41-50	F3SIG	Constant 3	Real
51-60	F4SIG	Constant 4	Real
61-70	F5SIG	Constant 5	Real

MEASURED SIGNAL

A measured signal makes available to the block diagram the present value of a referenced SAS4A/SASSYS-1 variable. The correspondence between the variable that is referenced and the signal card data field values is given in Table 2.7.6. Note that all measured signals have a JTYPE value in the range of -50 to -89 or -101 to -136, inclusive.

DEMAND SIGNAL

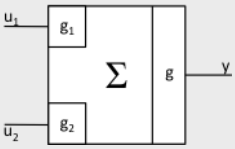
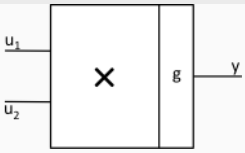
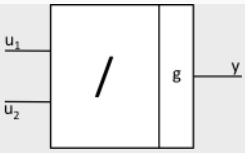
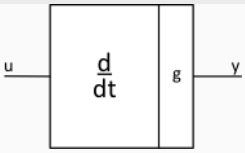
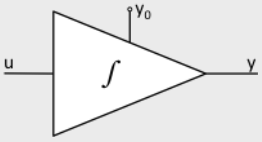
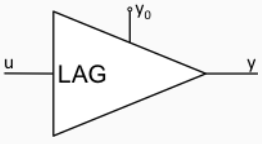
A demand signal makes available to the block diagram the product of the current value of a time dependent function defined by the user through a demand table and an initial condition value. A demand table is a set of ordered pair values supplied by the user in the format of Table 2.7.5 and described by *Data Card Input*.

The code obtains the demand signal by linearly interpolating among the table entries using the current time. The initial value is obtained as described in Section 6.4.3. The correspondence between the demand table and the signal card data fields is given in Table 2.7.6. Note that a demand signal has a JTYPE value of -90.

BLOCK SIGNAL

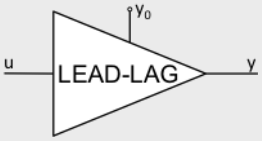
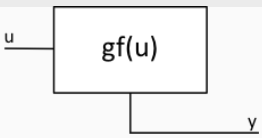
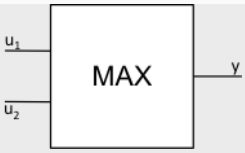
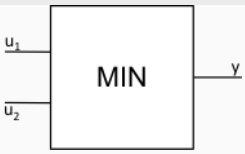
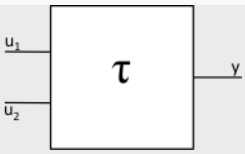
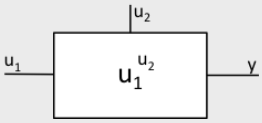
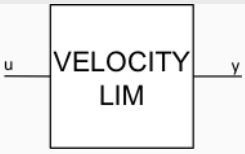
A block signal makes available to the block diagram the value at the output of a block. The correspondence between the block characteristics and the signal card data fields is given in Table 2.7.6. Note that all block signals have a JTYPE value in the range from 1 through 23. A measured, demand, or block signal can be used as an input to a block by specifying on the block's signal definition card the signal identification number assigned to the input signal. The signals input to each block type are combined according to the mathematical expressions given in Table 2.7.4.

Table 2.7.4: Basic Mathematical Blocks.

JTYPE	Block	Type	Representation	Mathematical Expression
1	summer	function		$y = g\left(g_1 u_1 + g_2 u_2\right)$
2	multiplier	function		$y = g u_1 u_2$
3	divider	function		$y = g\frac{u_1}{u_2}$
4	differentiator	function		$y = g\frac{\mathrm{d}}{\mathrm{dt}}u$
5	integrator	dynamic		$y = y_0 + g\int_0^t \mathrm{u}\mathrm{d}t'$
6	lag compensator	dynamic		$y + \tau\frac{\mathrm{d}}{\mathrm{dt}}y = gu$ $y\left(0\right) = y_0$

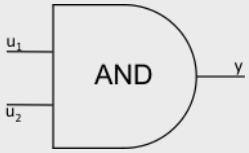
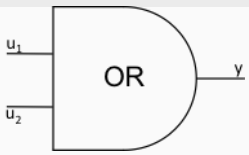
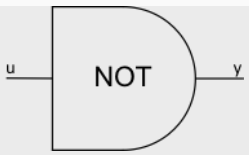
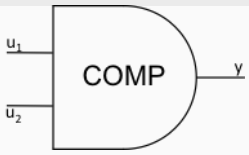
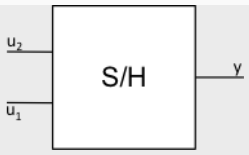
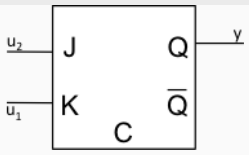
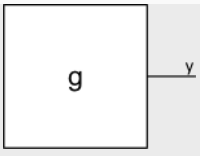
continues on next page

Table 2.7.4 – continued from previous page

JTYPE	Block	Type	Representation	Mathematical Expression
7	lead-lag compensator	dynamic		$y + \tau_1 \frac{\mathrm{d}}{\mathrm{dt}} y = g\left(u + \tau_2 \frac{\mathrm{d}}{\mathrm{dt}} u\right)$ $y(0) = y_0$
8	function generator	table		$y = gf(u)$
9	maximum value	function		$y = \max(u_1, u_2)$
10	minimum value	function		$y = \min(u_1, u_2)$
11	time delay	function		$y = y_0\ 0 \le t \le \tau$ $y = u(t - \tau)\ t \ge \tau$
12	natural logarithm	function		$y = \ln u$
13	exponential	function		$y = u_1^{u_2}$
14	velocity limiter	function		$y = y_{\mathrm{dn}}\ gu < y_{\mathrm{dn}}$ $y = y_{\mathrm{up}}\ gu > y_{\mathrm{up}}$ $y = gu$ otherwise $y_{\mathrm{dn}} = y(t - h) - hv_{\mathrm{dn}}$ $y_{\mathrm{up}} = y(t - h) + hv_{\mathrm{up}}$

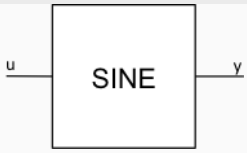
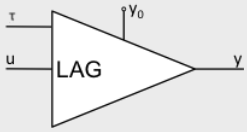
continues on next page

Table 2.7.4 – continued from previous page

JTYPE	Block	Type	Representation	Mathematical Expression
15	AND	logic		$y = 1\ u_1 > 0, u_2 > 0$ $y = 0$ otherwise
16	OR	logic		$y = 0\ u_1 \leq 0, u_2 \leq 0$ $y = 1$ otherwise
17	NOT	logic		$y = 0\ u \leq 0$ $y = 1\ u > 0$
18	comparator	logic		$y = 0\ u_1 < u_2$ $y = 1\ u_1 \geq u_2$
19	sample and hold	function		$y(t) = u_2(t)\ u_1(t) \leq 0$ $y(t) = u_2(t_0)\ u_1(t) \geq 0,$ $t \geq t_0$, where $u_1(t') \leq 0$ $t' < t_0$
20	J-K flip flop	logic		$y^{n+1} = Q^n\ u_1 \leq 0, u_2 \leq 0$ $y^{n+1} = 0\ u_1 > 0, u_2 \leq 0$ $y^{n+1} = 1\ u_1 \leq 0, u_2 > 0$ $y^{n+1} = \overline{Q}^n\ u_1 > 0, u_2 > 0$
21	constant	function		$y = g$

continues on next page

Table 2.7.4 – continued from previous page

JTYPE	Block	Type	Representation	Mathematical Expression
22	sine	function		$y = g_1 \sin(g_2 u + g_3)$
23	variable lag compensator	dynamic		$y + \tau(t)\frac{d}{dt}y = gu$ $y(0) = y_0$

CONTROL SIGNALS

A control signal is used to set the value of a SAS4A/SASSYS-1 variable equal to the value of a block signal. The correspondence between the block signal and the SAS4A/SASSYS-1 variable and the signal card data fields is given in Table 2.7.6. Note that all control signals have a JTYPE value that ranges from -1 through -10.

END OF SIGNALS

A sequence of signal definition cards is delimited by a signal card with the ISIG field entry equal to “999”.

This card also contains flags for the binary output file print interval and control of the steady-state solution finder. First, the absolute value of the JTYPE field for the 999 card is sets the print interval for control system results output to the binary output file CONTROL.dat. Second, the J1SIG field is used to determine whether the steady-state solution finder is to be used. An entry of “1” activates the steady-state solution finder, while any other entry in this field causes the solution finder to be bypassed. (A discussion of the initial condition option is given in Section 6.4.3.) Finally, the J2SIG field allows the user to control the amount of steady-state output generated. An entry of “1” produces an extended output for trouble shooting purposes, while any other entry produces a standard output.

The JTYPE field is also used to generate an extended print-out during the transient for debugging purposes. The debug print is generated by setting the JTYPE field of the 999 card to a negative value. The print-out begins at the time specified on the F1SIG field.

DATA CARDS

A data card contains the data fields for the FORTRAN variables LOC, N, VAR1, VAR2, VAR3, VAR4, and VAR5, with the record format (2I6,5E12.5). The variables are defined in *Data Card Input*.

A data card appearing in the control system block has a format identical to the standard SAS4A/SASSYS-1 data card used in other input blocks and is processed in the same way. The format information given above is the same as in the SAS manuals and is given here for completeness.

Data cards are used to construct demand tables and function generator tables, and to supply solution control parameters. These quantities and their storage locations are defined in *Data Card Input*.

Table 2.7.5: Data Card Format

Column	FORTRAN Symbol	Definition	Variable Type
1	LOC	Storage Location of VAR1	Integer
7	N	Number of Consecutive Locations	Integer
13	VAR1	Constant 1	Real
25	VAR2	Constant 2	Real
37	VAR3	Constant 3	Real
49	VAR4	Constant 4	Real
61	VAR5	Constant 5	Real

DATA CARD INPUT

1-2000 **CTLTAB(J,J1SIG)**

Table of normalized demand values. A table is defined when at least two entries are provided. Index J1SIG designates the table number and J is the entry number in the table.

Dimension (20,100).

2001-4000 **CTLTIM(J,J1SIG)**

Times for **CTLTAB** table. Values must be in ascending order.

Dimension (20,100).

4001-6000 **CTLFNC(J,J1SIG)**

Table of function generator dependent variables. A table is defined when at least two entries are provided. Index J1SIG designates the table number and J is the entry number in the table.

Dimension (20,100).

6001-8000 **CTLSIG(J,J1SIG)**

Table of independent variables for **CTLFNC** table. Values must be in ascending order.

Dimension (20,100).

8001 **EPSCS**

Convergence parameter for dynamic blocks over a subinterval. Must be greater than zero.

8002 **EPSCPL**

Maximum relative change in a control signal over a subinterval. Must be greater than zero.

SUMMARY OF CONTROL SYSTEM SIGNALS

Table 2.7.6: Signal Card Contents

Signal Type	Signal Variable	Card Contents
Measured	Compressible Volume Pressure, PRESL3	JTYPE = -50 J1SIG = Volume Number, ICV F4SIG = Y_0 (optional)
Measured	Liquid Segment Flowrate, FLOWSL3	JTYPE = -51 J1SIG = Liquid Segment Number, ISGL F4SIG = Y_0 (optional)
Measured	Liquid Cover Gas Interface Elevation, ZINTR3	JTYPE = -52 J1SIG = Volume Number, ICV F4SIG = Y_0 (optional)
Measured	Liquid Mass, XLQMS3	JTYPE = -53 J1SIG = Volume Number, ICV F4SIG = Y_0 (optional)
Measured	Cover Gas Volume, VOLGC3	JTYPE = -54 J1SIG = Volume Number, ICV F4SIG = Y_0 (optional)
Measured	Time	JTYPE = -55 F4SIG = Y_0 (optional)
Measured	Pump Head, HEADP3	JTYPE = -56 J1SIG = Pump Number, IPMP F4SIG = Y_0 (optional)
Measured	Liquid Temperature, TLQCV3	JTYPE = -57 J1SIG = Volume Number, ICV F4SIG = Y_0 (optional)
Measured	Liquid Density, DNSCV3	JTYPE = -58 J1SIG = Volume Number, ICV F4SIG = Y_0 (optional)
Measured	Wall Temperature, TWLCV3	JTYPE = -59 J1SIG = Volume Number, ICV J2SIG = Thick-walled node ID, optional F4SIG = Y_0 (optional)
Measured	Cover Gas Pressure, PRESG3	JTYPE = -60 J1SIG = Volume Number, ICV F4SIG = Y_0 (optional)
Measured	Cover Gas Mass, GASMS3	JTYPE = -61 J1SIG = Volume Number, ICV F4SIG = Y_0 (optional)
Measured	Cover Gas Temperature, TGASC3	JTYPE = -62 J1SIG = Volume Number, ICV F4SIG = Y_0 (optional)
Measured	Not Used.	JTYPE = -63
Measured	Liquid Segment Temperature, TSLIN3	JTYPE = -64 J1SIG = Segment Number, ISGL J2SIG = 1 (Inlet) J2SIG = 2 (Outlet) F4SIG = Y_0 (optional)

continues on next page

Table 2.7.6 – continued from previous page

Signal Type	Signal Variable	Card Contents
Measured	Pump Speed, PSPED3	JTYPE = -65 J1SIG = Pump Number, IPMP F4SIG = Y_0 (optional)
Measured	Core Channel Coolant Flowrate, CHFLO3	JTYPE = -66 J1SIG = Channel Number, ICH J2SIG = 1 (Inlet) J2SIG = 2 (Outlet) F4SIG = Y_0 (optional)
Measured	Liquid Node Temperature, TLNOD3	JTYPE = -67 J1SIG = Node Number, INOD F4SIG = Y_0 (optional)
Measured	Wall Node Temperature, TWNOD3	JTYPE = -68 J1SIG = Node Number, INOD F4SIG = Y_0 (optional)
Measured	Liquid Element Temperature, TELEM	JTYPE = -69 J1SIG = Element Number, IEL J2SIG = 1 (Inlet) J2SIG = 2 (Outlet) F4SIG = Y_0 (optional)
Measured	Not Used.	JTYPE = -70
Measured	Core Channel Outlet Temperature, CHFCOF	JTYPE = -71 J1SIG = Channel Number, ICH J2SIG = 1 (Inlet) J2SIG = 2 (Outlet) F4SIG = Y_0 (optional)
Measured	Normalized Reactor Power, EXP(POWVA(3,1))	JTYPE = -72 F4SIG = Y_0 (optional)
Measured	Normalized Fission Power, POWFS0*AMPO	JTYPE = -73 F4SIG = Y_0 (optional)
Measured	Normalized Decay Power, Sum(POWWT(I)*POWDKH(I))	JTYPE = -74 F4SIG = Y_0 (optional)
Measured	Equivalent Circuit EM Pump Voltage	JTYPE = -75 J1SIG = Pump Number, IPMP F4SIG = Y_0 (optional)
Measured	Equivalent Circuit EM Pump Frequency	JTYPE = -76 J1SIG = Pump Number, IPMP F4SIG = Y_0 (optional)
Measured	Equivalent Circuit EM Pump Current	JTYPE = -77 J1SIG = Pump Number, IPMP F4SIG = Y_0 (optional)
Measured	Equivalent Circuit EM Pump Phase Angle	JTYPE = -78 J1SIG = Pump Number, IPMP F4SIG = Y_0 (optional)
Measured	Not Used.	JTYPE = -79,…,-82
Measured	Steam Generator Feedwater Mass Inlet Flowrate	JTYPE = -83 J2SIG = SG Number F4SIG = Y_0 (optional)

continues on next page

Table 2.7.6 – continued from previous page

Signal Type	Signal Variable	Card Contents
Measured	Steam Generator Feedwater Inlet Enthalpy	JTYPE = -84 J2SIG = SG Number F4SIG = Y_0 (optional)
Measured	Steam Generator Steam Mass Flowrate	JTYPE = -85 J2SIG = SG Number F4SIG = Y_0 (optional)
Measured	Steam Generator Outlet Temperature	JTYPE = -86 J2SIG = SG Number F4SIG = Y_0 (optional)
Measured	Steam Generator Steam Pressure	JTYPE = -87 J2SIG = SG Number F4SIG = Y_0 (optional)
Measured	Steam Generator Water Level	JTYPE = -88 J2SIG = SG Number F4SIG = Y_0 (optional)
Measured	Steam Generator Steam Outlet Enthalpy	JTYPE = -89 J2SIG = SG Number F3SIG = Initial Condition Flag F4SIG = Y_o
Demand	Demand Table	JTYPE = -90 J1SIG = Demand Table Number J2SIG = Number of Entries in Table F4SIG = Y_o
Measured	Fuel Centerline Temperature	JTYPE = -101 J1SIG = Channel Number J2SIG = MZ Mesh Number F1SIG = Scaling Factor F2SIG = Offset F4SIG = Y_0 (optional)
Measured	Fuel Average Temperature	JTYPE = -102 J1SIG = Channel Number J2SIG = MZ Mesh Number F1SIG = Scaling Factor F2SIG = Offset F4SIG = Y_0 (optional)
Measured	Fuel Surface Temperature	JTYPE = -103 J1SIG = Channel Number J2SIG = MZ Mesh Number F1SIG = Scaling Factor F2SIG = Offset F4SIG = Y_0 (optional)
Measured	Clad Inner Wall Temperature	JTYPE = -104 J1SIG = Channel Number J2SIG = MZ Mesh Number F1SIG = Scaling Factor F2SIG = Offset F4SIG = Y_0 (optional)

continues on next page

Table 2.7.6 – continued from previous page

Signal Type	Signal Variable	Card Contents
Measured	Clad Mid Wall Temperature	JTYPE = -105 J1SIG = Chanel Number J2SIG = MZ Mesh Number F1SIG = Scaling Factor F2SIG = Offset F4SIG = Y_0 (optional)
Measured	Clad Outer Wall Temperature	JTYPE = -106 J1SIG = Channel Number J2SIG = MZ Mesh Number F1SIG = Scaling Factor F2SIG = Offset F4SIG = Y_0 (optional)
Measured	Coolant Temperature	JTYPE = -107 J1SIG = Channel Number J2SIG = MZC Mesh Number F1SIG = Scaling Factor F2SIG = Offset F4SIG = Y_0 (optional)
Measured	Coolant Pressure	JTYPE = -108 J1SIG = Channel Number J2SIG = MZC Mesh Number F1SIG = Scaling Factor F2SIG = Offset F4SIG = Y_0 (optional)
Measured	Coolant Saturation Temperature	JTYPE = -109 J1SIG = Channel Number J2SIG = MZC Mesh Number F1SIG = Scaling Factor F2SIG = Offset F4SIG = Y_0 (optional)
Measured	Coolant Boiling Margin	JTYPE = -110 J1SIG = Channel Number J2SIG = MZC Mesh Number F1SIG = Scaling Factor F2SIG = Offset F4SIG = Y_0 (optional)
Measured	Coolant Average Temperature	JTYPE = -111 J1SIG = Channel Number J2SIG = MZC Mesh Number F1SIG = Scaling Factor F2SIG = Offset F4SIG = Y_0 (optional)
Measured	Structure Inner Temperature	JTYPE = -112 J1SIG = Channel Number J2SIG = MZC Mesh Number F1SIG = Scaling Factor F2SIG = Offset F4SIG = Y_0 (optional)

continues on next page

Table 2.7.6 – continued from previous page

Signal Type	Signal Variable	Card Contents
Measured	Structure Outer Temperature	JTYPE = -113 J1SIG = Channel Number J2SIG = MZC Mesh Number F1SIG = Scaling Factor F2SIG = Offset F4SIG = Y_0 (optional)
Measured	Reflector Inner Temperature	JTYPE = -114 J1SIG = Channel Number J2SIG = MZC Mesh Number F1SIG = Scaling Factor F2SIG = Offset F4SIG = Y_0 (optional)
Measured	Reflector Outer Temperature	JTYPE = -115 J1SIG = Channel Number J2SIG = MZC Mesh Number F1SIG = Scaling Factor F2SIG = Offset F4SIG = Y_0 (optional)
Measured	Peak Fuel Temperature	JTYPE = -116 J1SIG = Channel Number J2SIG = (Optional) If non-zero, peak fuel temperature is provided over all channels in the range J1SIG ≤ ICH ≤ J2SIG F1SIG = Scaling Factor F2SIG = Offset F4SIG = Y_0 (optional)
Measured	Peak Clad Temperature	JTYPE = -117 J1SIG = Channel Number J2SIG = (Optional) If non-zero, peak cladding temperature is provided over all channels in the range J1SIG ≤ ICH ≤ J2SIG F1SIG = Scaling Factor F2SIG = Offset F4SIG = Y_0 (optional)
Measured	Peak Coolant Temperature	JTYPE = -118 J1SIG = Channel Number J2SIG = (Optional) If non-zero, peak coolant temperature is provided over all channels in the range J1SIG ≤ ICH ≤ J2SIG F1SIG = Scaling Factor F2SIG = Offset F4SIG = Y_0 (optional)

continues on next page

Table 2.7.6 – continued from previous page

Signal Type	Signal Variable	Card Contents
Measured	Minimum Boiling Margin	JTYPE = -119 J1SIG = Channel Number J2SIG = (Optional) If non-zero, minimum boiling margin temperature is provided over all channels in the range J1SIG ≤ ICH ≤ J2SIG F1SIG = Scaling Factor F2SIG = Offset F4SIG = Y_0 (optional)
Measured	Coolant Inlet Temperature	JTYPE = -120 J1SIG = Channel Number F1SIG = Scaling Factor F2SIG = Offset F4SIG = Y_0 (optional)
Measured	Coolant Inlet Pressure	JTYPE = -121 J1SIG = Channel Number F1SIG = Scaling Factor F2SIG = Offset F4SIG = Y_0 (optional)
Measured	Coolant Inlet Flowrate	JTYPE = -122 J1SIG = Channel Number F1SIG = Scaling Factor F2SIG = Offset F4SIG = Y_0 (optional)
Measured	Coolant Outlet Temperature	JTYPE = -123 J1SIG = Channel Number F1SIG = Scaling Factor F2SIG = Offset F4SIG = Y_0 (optional)
Measured	Coolant Outlet Pressure	JTYPE = -124 J1SIG = Channel Number F1SIG = Scaling Factor F2SIG = Offset F4SIG = Y_0 (optional)
Measured	Coolant Outlet Flowrate	JTYPE = -125 J1SIG = Channel Number F1SIG = Scaling Factor F2SIG = Offset F4SIG = Y_0 (optional)
Measured	Maximum Coolant Outlet Temperature	JTYPE = -126 F1SIG = Scaling Factor F2SIG = Offset F4SIG = Y_0 (optional)
Measured	Minimum Coolant Outlet Temperature	JTYPE = -127 F1SIG = Scaling Factor F2SIG = Offset F4SIG = Y_0 (optional)

continues on next page

Table 2.7.6 – continued from previous page

Signal Type	Signal Variable	Card Contents
Measured	Pin Bundle ΔT	JTYPE = -128 J1SIG = Channel Number F1SIG = Scaling Factor F2SIG = Offset F4SIG = Y_0 (optional)
Measured	Pin Bundle ΔP	JTYPE = -129 J1SIG = Channel Number F1SIG = Scaling Factor F2SIG = Offset F4SIG = Y_0 (optional)
Measured	Assembly Bundle ΔT	JTYPE = -130 J1SIG = Channel Number F1SIG = Scaling Factor F2SIG = Offset F4SIG = Y_0 (optional)
Measured	Assembly Bundle ΔP	JTYPE = -131 J1SIG = Channel Number F1SIG = Scaling Factor F2SIG = Offset F4SIG = Y_0 (optional)
Measured	Assembly Power	JTYPE = -132 J1SIG = Channel Number F1SIG = Scaling Factor F2SIG = Offset F4SIG = Y_0 (optional)
Measured	Pin Linear Power	JTYPE = -133 J1SIG = Channel Number J2SIG = MZ Mesh Number F1SIG = Scaling Factor F2SIG = Offset F4SIG = Y_0 (optional)
Measured	Peak Pin Linear Power	JTYPE = -134 J1SIG = Channel Number F1SIG = Scaling Factor F2SIG = Offset F4SIG = Y_0 (optional)
Measured	Fission Gas Plenum Temperature	JTYPE = -135 J1SIG = Channel Number F1SIG = Scaling Factor F2SIG = Offset F4SIG = Y_0 (optional)
Measured	Fission Gas Plenum Pressure	JTYPE = -136 J1SIG = Channel Number F1SIG = Scaling Factor F2SIG = Offset F4SIG = Y_0 (optional)

continues on next page

Table 2.7.6 – continued from previous page

Signal Type	Signal Variable	Card Contents
Block	Summer	JTYPE = 1 J1SIG = Input Signal 1, ISIG J2SIG = Input Signal 2, ISIG F1SIG = g_1 F2SIG = g_2 F3SIG = g F4SIG = Y_0 (optional)
Block	Multiplier	JTYPE = 2 J1SIG = Input Signal 1, ISIG J2SIG = Input Signal 2, ISIG F1SIG = g F4SIG = Y_0 (optional)
Block	Divider	JTYPE = 3 J1SIG = Input Signal 1, ISIG J2SIG = Input Signal 2, ISIG F1SIG = g F4SIG = Y_0 (optional)
Block	Differentiator	JTYPE = 4 J1SIG = Input Signal 1, ISIG F1SIG = g F4SIG = Y_0 (optional)
Block	Integrator	JTYPE = 5 J1SIG = Input Signal 1, ISIG F1SIG = g F3SIG = Initial Condition Flag F4SIG = Y_0 F5SIG = E_z (Zero Crossing Parameter)
Block	Lag Compensator	JTYPE = 6 J1SIG = Input Signal 1, ISIG F1SIG = g F2SIG = Tau F4SIG = Y_0 (Not used when J1SIG(999) = 1) F5SIG = E_z (Zero Crossing Parameter)
Block	Lead - Lag Compensator	JTYPE = 7 J1SIG = Input Signal 1, ISIG F1SIG = g F2SIG = τ_1 F3SIG = τ_2 F4SIG = $Y_0 Y_o$ (Not used when J1SIG(999) = 1) F5SIG = E_z (Zero Crossing Parameter)
Block	Function Generator	JTYPE = 8 J1SIG = Input Signal 1, ISIG J2SIG = Function Generator Table Number F1SIG = g F4SIG = Y_0 (optional)

continues on next page

Table 2.7.6 – continued from previous page

Signal Type	Signal Variable	Card Contents
Block	Maximum	JTYPE = 9 J1SIG = Input Signal 1, ISIG J2SIG = Input Signal 2, ISIG F4SIG = Y_0 (optional)
Block	Minimum	JTYPE = 10 J1SIG = Input Signal 1, ISIG J2SIG = Input Signal 2, ISIG F4SIG = Y_0 (optional)
Block	Time Delay	JTYPE = 11 J1SIG = Input Signal 1, ISIG F1SIG = τ F4SIG = Y_0 (Not used when J1SIG(999) = 1)
Block	Natural Logarithm	JTYPE = 12 J1SIG = Input Signal 1, ISIG F1SIG = g F4SIG = Y_0 (optional)
Block	Exponentation	JTYPE = 13 J1SIG = Input Signal 1, ISIG J2SIG = Input Signal 2, ISIG F1SIG = g F4SIG = Y_0 (optional)
Block	Velocity Limiter	JTYPE = 14 J1SIG = Input Signal 1, ISIG F1SIG = V_{down} F2SIG = V_{up} F3SIG = g F4SIG = Y_0 (optional)
Block	AND	JTYPE = 15 J1SIG = Input Signal 1, ISIG J2SIG = Input Signal 2, ISIG F4SIG = Y_0 (optional)
Block	OR	JTYPE = 16 J1SIG = Input Signal 1, ISIG J2SIG = Input Signal 2, ISIG F4SIG = Y_0 (optional)
Block	NOT	JTYPE = 17 J1SIG = Input Signal 1, ISIG F4SIG = Y_0 (optional)
Block	Comparator	JTYPE = 18 J1SIG = Input Signal 1, ISIG J2SIG = Input Signal 2, ISIG F4SIG = Y_0 (optional)
Block	Sample and Hold	JTYPE = 19 J1SIG = Input Signal 1, ISIG J2SIG = Input Signal 2, ISIG F4SIG = Y_0 (optional)

continues on next page

Table 2.7.6 – continued from previous page

Signal Type	Signal Variable	Card Contents
Block	JK Flip-Flop	JTYPE = 20 J1SIG = Input Signal 1, ISIG J2SIG = Input Signal 2, ISIG F4SIG = Q_0
Block	Constant	JTYPE = 21 F1SIG = g F4SIG = Y_0 (optional)
Block	Sine Function	JTYPE = 22 J1SIG = Input Signal 1, ISIG F1SIG = g1 F2SIG = g2 F3SIG = g3 F4SIG = Y_0 (optional)
Block	Variable Lag Compensator	JTYPE = 23 J1SIG = Input Signal 1, ISIG (u) J2SIG = Input Signal 2, ISIG (Tau) F1SIG = g F4SIG = Y_0 (Not used when J1SIG(999) = 1) F5SIG = E_z (Zero Crossing Parameter)
Control	Reactivity, $	JTYPE = -1 J1SIG = Signal Number Used F4SIG = Y_0 (optional) F5SIG = E_z (Zero Crossing Parameter)
Control	Normalized Pump Motor Torque	JTYPE = -2 J1SIG = Signal Number Used J2SIG = Pump Number F4SIG = Y_0 (optional) F5SIG = E_z (Zero Crossing Parameter)
Control	Steam Generator Feedwater Mass Flowrate	JTYPE = -3 J1SIG = Signal Number Used J2SIG = Steam Generator Number F4SIG = Y_0 (optional) F5SIG = E_z (Zero Crossing Parameter)
Control	Steam Generator Feedwater Enthalpy	JTYPE = -4 J1SIG = Signal Number Used J2SIG = Steam Generator Number F4SIG = Y_0 (optional) F5SIG = E_z (Zero Crossing Parameter)
Control	Steam Generator Steam Mass Flowrate	JTYPE = -5 J1SIG = Signal Number Used J2SIG = Steam Generator Number F4SIG = Y_0 (optional) F5SIG = E_z (Zero Crossing Parameter)
Control	Sodium Valve Loss Coefficient	JTYPE = -6 J1SIG = Signal Number Used J2SIG = Valve Number F4SIG = Y_0 (optional) F5SIG = E_z (Zero Crossing Parameter)

continues on next page

Table 2.7.6 – continued from previous page

Signal Type	Signal Variable	Card Contents
Control	Steam Generator Steam Pressure	JTYPE = -7 J1SIG = Signal Number Used J2SIG = Steam Generator Number F4SIG = Y_0 (optional) F5SIG = E_z (Zero Crossing Parameter)
Control	Air Dump Heat Exchanger Air Mass Flowrate	JTYPE = -8 J1SIG = Signal Number Used J2SIG = Air Dump Heat Exchanger Number F4SIG = Y_0 (optional) F5SIG = E_z (Zero Crossing Parameter)
Control	*Simulation Trip*	JTYPE = -10 J1SIG = Signal Number Used J2SIG = Latching option (0, Unlatched; 1, Latched) F4SIG = Y_0 (optional) F5SIG = E_z (Zero Crossing Parameter)

2.7.4 Block 6 — IINBOP — Balance-of-Plant Integer Input

Note: The first-encountered IINBOP is saved and may not be redefined by subsequent data. Within the block, the data must be presented as defined in ANL/RAS 89/6. This block must precede Block 15, *FINBOP*.

See ANL/RAS 89/6, Appendix B.

The input format for each record in Block 6 is:

```
(2I3,11I6)
```

The first entry on a line identifies the type of geometric components for which data are being entered. The categories of components are numbered as follows:

Category	Description
1	compressible volume
2	segment
3	element
4	steam generator
5	waterside pump
6	volume boundary condition
7	not currently used
8	supersegment
9	flags to invoke options
10	standard valve
11	balance-of-plant legs
12	check valve
13	selection of parameters for printing
14	relief valve.

One point about these categories needs clarification. There is a category for general elements (category 3), and there are also categories for specific types of elements (pumps (5), standard valves (10), and check valves(12)). The data which fall under the general elements category must be entered for all elements. In addition, some element types require data unique to that type of element; these are entered under the appropriate element category. For example, the length of an element must be entered for all elements and so appears in the general elements category, whereas pump motor torque must be entered only for pumps and so appears in the pump elements category. Therefore, some information about a pump is entered in the general elements category and the rest in the pump elements category.

The second entry is the continuation card number. This accommodates components which have more fixed point input entries than can fit on one card. The continuation card number is 1 for the first fixed-point data card for a component, 2 for the second card, etc. The third entry gives the component number in the user's nodalization. The numbering of steam generators must be the same on the sodium side and on the water side. Pumps and valves on the water side are numbered independently from pumps and valves on the sodium side. The remaining data entries vary with the geometric component and are as follows:

Compressible Volume

(2I3,11I6): 1, 1, user's no., NTPCVW, NCVBCW, NSUPSG, NQFLG. NLGCVW, NENTRF

NTPCVW: A volume can be filled with single-phase liquid, single-phase vapor, or two-phase fluid. In addition, the pseudo-volume which marks the subcooled/two-phase interface in the evaporator is treated as a special case. NTPCVW is used to distinguish these four categories of volumes, with

= 1 for a single-phase liquid volume,
= 2 for a single-phase vapor volume,
= 3 for a two-phase volume;
= 4 for the pseudo-volume at the liquid/two-phase interface in the evaporator.

NCVBCW: Volumes which perform certain functions (i.e., heater volumes, steam generator outlet plena) must be flagged, and NCVBCW is used to flag them as follows:

= 0 for a standard volume,
= 1 for a volume boundary condition volume,
= 2 for an inlet flow boundary condition volume,
= 3 for an outlet flow boundary condition volume,
= 4 for a steam generator inlet plenum,
= 5 for a steam generator outlet plenum,
= 6 for a heater volume,
= 7 for a turbine.

The designation "standard volume" simply means any volume which does not fall into one of the categories for NCVBCW = 1 through 7.

NSUPSG: If the volume is contained within a supersegment, NSUPSG must be entered and given the number of the supersegment. NSUPSG is not entered for volumes which begin or end a supersegment.

NQFLG flags whether or not a compressible volume is a heater, with NQFLG:

= 0 if the volume is not a heater and
= the user's number for the heater if the volume is a heater.

NLGCVW is the number of the leg of the nodalization to which the volume belongs (See ANL/RAS 89/6, Section 8.3 for an explanation of how a nodalization is divided into legs).

NENTRF: The thermodynamic state of a compressible volume can be specified through floating point input data in several ways. The user sets NENTRF for each volume to tell the code which thermodynamic quantities are being entered for that volume, with

= 1, single-phase volumes; pressure and temperature entered,
= 2, single-phase volumes; pressure and enthalpy entered,
= 3, two-phase volumes; pressure and quality entered,
= 4, two-phase volumes; temperature and quality entered,
= 5, two-phase heater volumes; pressure, two-phase level, and ambient temperature entered,
= 6, two-phase heater volumes; temperature, two-phase level, and ambient temperature entered.

If the volume is attached to a flow boundary condition, the following additional quantities are entered:

NBCINF, NFLSEG, IFBWCL

NBCINF: the number of the table in which boundary condition data are stored if the boundary condition is controlled through a user-input table, rather than by the control system.

NFLSEG: If the boundary condition is controlled by data from a table input by the user, the user must choose which thermodynamic data to enter and must signal this choice to the code through the flag NFLSEG, with

= 0, if enthalpy is entered,
= 1, if temperature and pressure are entered for a subcooled liquid boundary condition,
= 2, if temperature and pressure are entered for a superheated steam boundary condition,
= 3, if quality and pressure are entered for a two-phase boundary condition,
= 4, if quality and temperature are entered for a two-phase boundary condition.

For an outflow boundary condition, NFLSEG = -1.

IFBWCL: flags whether the boundary condition is controlled by a table or by the control system, with

= 0, if the boundary condition is controlled by a table,
= 1, if the boundary condition is controlled by the control system.

If the volume is a volume boundary condition, the following additional quantities are entered:

NBCINP, IVBWCL

NBCINP is the number of the table in which boundary condition data are stored if the boundary condition is controlled through a user-input table, rather than by the control system,

IVBWCL flags whether the boundary condition is controlled by a table or by the control system, with

= 0, if the boundary condition is controlled by a table,
= 1, if the boundary condition is controlled by the control system.

Segment

(2I3,11I6): 2, 1, user's no., JCVW(1), JCVW(2), NODMAX

JCVW(1) is the compressible volume number in the user's nodalization for the volume at the segment inlet,

JCVW(2) is the user's volume number at the segment outlet,

NODMAX is the maximum number of enthalpy transport nodes into which may be tracked along a segment. See ANL/RAS 89/6, Section 4.4.2, for a discussion of the enthalpy transport model.

(2I3,11I6): 2, 2, user's no., JCV1FG, IHTSEG, IHTLW, IHTUP

This input line is used only for heaters using a heater model other than the simple heater model.

JCV1FG indicates where a segment attached to a heater volume is attached to the volume, with

= -1, if the segment is attached to the bottom of the volume,
= 0, if the segment is attached in between the top and the bottom of the volume,
= 1, if the segment is attached to the top of the volume.

IHTSEG is the user's number of the heater volume through which the segment passes,

IHTLW is entered if the segment is attached to a drain and is the user's number of the volume containing the drain,

IHTUP is entered if the segment is attached to a desuperheating section and is the user's number of the volume containing the desuperheating section.

Element

(2I3,11I6): 3, 1, user's no., NBOREL(1), NBOREL(2), NELSGW, ITYPW

NBOREL(1) is the element number of the upstream neighboring element in the user's nodalization. The code uses the convention that the first element in a segment is the one furthest upstream, and for this element, NBOREL(1) is set to 0.

NBOREL(2) is the element number of the downstream neighboring element in the user's nodalization. The code uses the convention that the last element in a segment is the one furthest downstream, and for this element, NBOREL(2) is set to -1.

NELSGW is the user's number of the segment in which the element is located.

ITYPW identifies the element type, with

= 3, for a pipe,
= 4, for a check valve,
= 5, for a pump,
= 6, for a heated element,
= 7, for a nozzle,
= 8, for a superheater,
= 11, for a valve.

(These element types use the same numbering as those on the sodium side wherever possible).

Steam Generator

a) IEVAP(ISGN) = 1 (evaporator) or 3 (once-through)

(2I3,11I6): 4, 1, steam gen. number, ICVSGN(1), ICVSGN(2), NOSGW, NODSC, NODTP, NODSH, IDUM1, IDUM2, LMPDOT

ICVSGN(1) is the user's number for the compressible volume which serves as the steam generator inlet plenum,

ICVSGN(2) is the user's number for the volume which is the outlet plenum,

NOSGW is the user's number for the segment which is at the outlet of the vapor leg which is fed by the steam generator (used for saving plot data only),

NODSC is the number of nodes in the subcooled zone,

NODTP is the number of nodes in the two-phase zone,

NODSH is the number of nodes in the superheated zone,

IDUM1 is a dummy integer,

IDUM2 is a dummy integer,

LMPDOT is the number of time steps used to compute an average value for PDOT, the derivative of pressure with respect to time.

(2I3,11I6): 4, 2, steam gen. number, ISGBUG, IHELE, IOPT1, IOPT2

ISGBUG is the number of PRIMAR time steps between debug prints for the steam generator calculation. If 0 is entered, no debug prints will be generated.

IHELE is an indicator for the geometry in the evaporator/steam generator: = 0 for straight tube, = 1 for helical coil.

IOPT1 is an indicator for the search option in the subcooled zone: = 1 on calibration factor, = 2 on length.

IOPT2 is an indicator for the search option in the superheated zone: = 1 on calibration factor, = 2 on length.

b) IEVAP(ISGN) = 2 (superheater)

(2I3,11I6): 4, 1, steam gen. number, NELSUH, NODSHT

NELSUH is the user's element number for the superheater associated with the evaporator (if any),

NODSHT is the number of nodes in the superheater.

(2I3,11I6): 4, 2, steam gen. number, ISHBUG, IHELS

ISHBUG is reserved.

IHELS is an indicator for the geometry in the superheater: = 0 for straight tube, = 1 for helical coil.

See ANL/RAS 90/1 for a detailed explanation of these steam generator parameters. Variable **IEVAP** should be set to 2 if a superheater is used, and to 3 if only an evaporator is used.

Pump

(2I3,11I6): 5, 1, pump number, IEMPW, IELPW, ILRPW, IPMWCL

IEMPW is the number designating the type of pump model chosen, i.e.,

= 0, if a table of pump head vs. time is entered,

= 1, if centrifugal pump model 1 is used,
= 2, if centrifugal pump model 2 is used.

Identical pump models are used for both waterside and sodium pumps, and so the discussion in ANL/RAS 84-14 of sodium pump models is the best source of more detailed information about the waterside pump models.

IELPW is the user's element number of the pump,

ILRPW is the flag which activates the locked rotor modeling, i.e.

= 1, for a locked rotor, with pump speed set to 0, and
= -2, if a table of pump head vs. flow is entered.

Unless a table of pump head vs. flow is entered, ILRPW is initially set to 0; if the code computes that the flow or pump speed becomes so low as to lock the rotor, ILRPW will automatically be reset to 1 by the code.

IPMWCL is the flag which routes control of the pump to the control system, with

= 0, for table lookup of the pump motor torque,
= 1, for control of the pump motor torque by the control system

Volume boundary condition

(2I3,11I6): 6, 1, table number, NTABVL

The thermodynamic state of a volume boundary condition volume can be specified by any of six choices of input data to be entered in the floating point volume boundary condition table. The code determines which choice the user has made from the variable NTABVL, with

= 1, for pressure and enthalpy entered for a liquid volume,
= 2, for pressure and temperature entered for a liquid volume,
= 3, for pressure and enthalpy entered for a vapor volume,
= 4, for pressure and temperature entered for a vapor volume,
= 5, for pressure and quality entered for a two-phase volume,
= 6, for temperature and quality entered for a two-phase volume.

Supersegment

(2I3,11I6): 8, 1, superseg. number, NSSIN, NSSOUT

NSSIN is the compressible volume number at the supersegment inlet,

NSSOUT is the volume number at the supersegment outlet.

See ANL/RAS 89/6, Secton 8.2, for an explanation of how a supersegment is set up.

Option flags

(2I3,11I6): 9, 1, NTRNPT, IBOPRT

NTRNPT flags whether or not enthalpy transport is used in vapor-filled segments, with

= 0, if enthalpy transport is used,
= 1, if enthalpy transport is not used.

See ANL/RAS 89/6, Section 4.4.2, for a detailed explanation of when the enthalpy transport model should or should not be used in segments composed of superheated vapor.

IBOPRT is the number of PRIMAR time steps between full prints of the balance-of-plant parameters. If IBOPRT is not entered, the code sets IBOPRT to 1.

Standard valve

(2I3,11I6): 10 1 user's number IVLELW IVLWCL

IVLELW is the user's number for the element which contains the valve,

IVLWCL flags whether the valve is controlled by a table or by the control system. If the valve is controlled by the control system, there are two options: 1) have the control system specify the valve driving function as a function of time, or 2) have the control system specify the valve stem position as a function of time. The code uses IVLWCL to determine which of these three choices the user has made, with IVLWCL

= 0 if the valve is controlled by a table,
= 1 if the driving function VFRACL is specified by the control system,
= 2 if the stem position VSTEMW is specified by the control system.

See Sec. 4.2 for a detailed description of the valve model.

Balance-of-plant legs

(2I3,11I6): 11, 1, LEGORD

LEGORD orders the legs of the balance of plant for the purpose of printing output (See ANL/RAS 89/6, Section 8.3, for a discussion of how the balance of plant can be divided into legs).

Check valve

(2I3,11I6): ,12, 1, user's number, ICVLEW, ICHVLK(1), ICHVLK(2), NCHVST

ICVLEW is the user's number for the element containing the valve,

ICHVLK(1) specifies the type of valve closure criterion entered. The user must choose between having the valve begin to close when the pressure drop across the valve falls below a user-input number or having it begin to close when the flow through the valve becomes less than a user-input number. This choice is communicated to the code by ICHVLK(1), with

= 1, if the valve begins to close when a pressure drop criterion is satisfied,
= 2, if the valve begins to close when a flow criterion is satisfied.

ICHVLK(2) specifies the type of valve opening criterion entered. The user must make the same choice as for valve closure,and this choice is designated through ICHVLK(2), with

= 1, if the valve begins to open when a pressure drop criterion is satisfied,
= 2 if the valve begins to open when a flow criterion is satisfied.

NCHVST specifies the current state of the valve, with

= 1, if the valve is fully open and will begin to close if the pressure drop across the valve is less than the user-input value CHEPS1,
= 2, if the valve is fully open and will begin to close if the flow through the valve is less than CHEPS1,
= 3, if the valve is in the process of closing,
= 4, if the valve is closed and will begin to open if the pressure drop across the valve exceeds the user-input value CHEPS2,
= 5, if the valve is closed and will begin to open if the flow through the valve exceeds CHEPS2,
= 6 if the valve is in the process of opening.

The initial value of NCHVST should always be entered as either 1, 2, 4, or 5; as the transient progresses and the valve opens and/or closes, the code will update NCHVST to reflect the current state of the valve.

Selection of parameters for printing

(2I3,11I6): 13, 1, JPRINT(17)

The user may select some or all of 17 parameters to be printed by setting the appropriate JPRINT array element to 1. The JPRINT array is ordered as follows:

JPRINT(1) all compressible volume pressures

JPRINT(2) steam generator subcooled/two-phase interface pressures

JPRINT(3) flows in all segments except flow boundary conditions and evaporator subcooled regions

JPRINT(4) flows in all standard valves only

JPRINT(5) flows in all pumps only

JPRINT(6) flows in all evaporator subcooled regions only

JPRINT(7) all evaporator outlet flows only

JPRINT(8) flows at all flow boundary conditions

JPRINT(9) mixture enthalpies in all compressible volumes

JPRINT(10) temperatures in all compressible volumes

JPRINT(11) densities in all compressible volumes

JPRINT(12) outlet enthalpies for all elements

JPRINT(13) outlet pressures for all elements

JPRINT(14) orifice coefficients for all elements

JPRINT(15) orifice coefficients for all standard valves only

JPRINT(16) pump head for all pumps

JPRINT(17) pump speed for all pumps.

If JPRINT(1) is set to 2, all 17 prints will be made.

Relief valve

(2I3,11I6): 14, 1, user's number, IRVLVW

IRVLVW is the user's number for the element assigned to the check valve.

2.7.5 Block 11 — OPCIN — Channel-Independent Floating-Point Input

1 EPSTEM K

Steady-state temperature convergence delta. The steady-state temperatures in the fuel-pin must converge such that |T' - T| <= **EPSTEM**, where T' is the temperature on the last iteration and T is the temperature on the previous iteration.

Suggested value: 0.1.

2 EPSREA

Not currently used.

3 EPSPOW

Neutron flux amplitude convergence criterion.

4 EPSGAM

Not currently used.

5 DT0 s

Initial and maximum main time step length.

6 DTMXB s

Maximum heat-transfer time step length after coolant boiling inception.

Suggested value: 0.01 s.

Maximum value: 0.02 s.

7 TIMAX s

Maximum problem time. Restart file is saved when time **TIMAX** is reached if **NSTEP**> 0.

8 REAITR

Not currently used.

9 TCOSTP s

Number of CPU seconds reserved at end of run for writing of restart file when **NSTEP** > 0.

10 **DTFUEL** K

Maximum fuel temperature change per heat-transfer time step.

Suggested value: 50.0 for **IMETAL** = 0,

5.0 for **IMETAL** > 0.

11 **DTCLAD** K

Maximum cladding temperature change per heat-transfer time step.

Suggested value: 30.0.

12 **TCLMAX**

Not currently used.

13 **DTP0** s

Initial PRIMAR time step size.

14 **DTPMAX** s

Maximum PRIMAR time step size before boiling starts.

15 **DTPBOI** s

Maximum PRIMAR step size after start of boiling.

16 **DTPLEV** s

Maximum PRIMAR step size after LEVITATE starts.

17 **DTPFCI** s

Maximum PRIMAR step size after PLUTO-2 starts.

18 **TPRFCI** s

Time after PLUTO-2 initiation when PRIMAR step can be increased to **DTPBOI**.

19 **DPINMX** Pa

Maximum change in inlet pressure per PRIMAR step.

20 **DTINMX** K

Maximum change in inlet temperature per PRIMAR step.

21 **DTMMXB** s

Maximum main time step after the onset of boiling.

22 **DPWMAX**

Maximum fractional change in power per main time step, only used if **IPOWER**= 1.

23 **ASCRAM**

Not currently used.

24 **PSCRAM**

Normalized total power for scram initiation.

Relevant for **IPOWER**=0, **NPK**=0.

25 **GSCRAM**

Not currently used.

26 **DTSSCP** s

Time step size for the null transient to initialize component-component heat transfer in PRIMAR-4.

27 **EPSSCP**

Component-component heat transfer convergence criterion for the null transient.

28-87 **DXMAPR**

Not currently used.

88 **DTNULL** s

Time step size for the null transient for steady-state core channel thermal hydraulics.

Not used if **ISSNUL**=0.

89 **TPCOST** s

Time to start writing PINACLE data for core image (PNPICO subroutine).

90 **TPCOLE** s

Time to start writing LEVITATE data for core image (LEPICO subroutine).

91 **TPCOEN** S

Time to stop writing data for core images.

92 **DTPICP** S

Time interval for writing data for core images while PINACLE is active.

93 **DTPICL** S

Time interval for writing data for core images while LEVITATE is active.

94 **DTMMAX** S

Maximum main time step length. Default = **DT0**.

95-104 **DTMIN (I)** S

Main time step table. **DTMIN**(I) is the length of the main time step to be used in the problem time interval from **TDTMIN**(I) to **TDTMIN**(I+1). For times before **TDTMIN**(1), **DT0** is used as the main time step. If J is the index of the last entry in the **DTMIN** table, then the main time step for times after **TDTMIN**(J) is **DTMIN**(J). If there are no entries in the **DTMIN** table, the automatic main time step selection algorithm will be used.

105-114 **TDTMIN (I)** S

See **DTMIN** above.

115-149 **TMAINS** S

Problem times for forced main time step termination.

150 **DELSCR** S

Time delay after scram initiation (**ASCRAM**, **PSCRAM**, **GSCRAM**) for scram reactivity table (**SCRTAB**, **SCRTME**) application and for external source (**KQSCRA**) changes.

151 **DELPUM** S

Time delay after scram initiation (**ASCRAM**, **PSCRAM**, **GSCRAM**) for beginning of **PUHALF** pump coastdown.

152 **PUHALF** S

Flow halving time after TSCRAM+DELPUM.

F(t)=F(0)/(1+t/**PUHALF**), where t is the time after TSCRAM+DELPUM. Only for **IFLOW** > 0.

153 **OPCDUM**

Not currently used.

154 **GRAVITY** **m/s^2**

Acceleration of gravity. Default: 9.81 m/s^2

155 **ADJKTOL**

Steady state initialization adjustment tolerance for internal loss coefficients. Internal loss coefficients are defined in **XKORI** for core channels and **G2PRDR** for elements. Must be greater than or equal to zero. Default: 0.0

When 0.0, the adjustment is unconstrained.

156-250 **OPCDUM**

Not currently used.

DEFAULT VALUES

DTP0	1.0
DTPBOI	0.01
DTPLEV	0.004
DTPFCI	0.001
TPRFCI	0.05
DPINMX	25000.0
DTINMX	5.0
DTPMAX	1.0
DTMMXB	0.02

2.7.6 Block 12 — POWINA — Power, Decay-Heat, and Reactivity Input

1 **POW** **W**

Steady-state power in the peak axial fuel pin segment. (sum of fuel + cladding + coolant + structure power; See IPOWOP). Note: **POW**=**POWTOT***FRPR/POWSUM where POWSUM=sum over **NCHAN** channels of (**NPIN***NSUBAS*(sum over MZ of **PSHAPE**)). (See MZ, **NCHAN**, **PSHAPE**, **NPIN**, **FRPR**, and **NSUBAS**).

2 **GENTIM** **s**

Prompt neutron lifetime.

3 **POWTOT** **W**

Total reactor power.

4-9 **BETADN (L)**

Effective delayed neutron fraction for delayed neutron precursor family L. L =1,…,**NDELAY**.

10-15 **DECCON (L)** **1/s**

Decay constant for delayed neutron precursor family L. L =1, …, **NDELAY**.

16-21 **OLDBDK (L)**

Not currently used.

22-27 **OLDDKL (L)**

Not currently used.

28 **OLDBDT**

Not currently used.

29-48 **PREATB (L)**

Transient external reactivity or power table utilized by function PREA for **NPREAT** > 0. If reactivity is input, entries are in dollars. If power is input, entries are normalized to nominal power (so that for **PREATM** = 0.0, **PREATB** = 1.0). L = 1, …, **NPREAT**.

49-68 **PREATM (L)** **s**

Transient problem times at which values in **PREATB** table are to be applied. L = 1, …, **NPREAT**.

69 **FRPR**

Fraction of total reactor power represented by sum of all channels.

70 **FRFLOW**

Fraction of total reactor coolant flow represented by sum of all channels.

71 **CRDLEN** **m**

Length of control rod drives washed by outlet sodium, for **ICREXP** = 1 (single node model). Typical value: 6 m.

72 **CRDEXP** **1/K**

Thermal expansion coefficient of control rod drives. Typical value: 2×10^{-5}.

73 **ACRDEX** **$/m**

Control rod expansion feedback = **ACRDEX** * DZ + **BCRDEX** * DZ^2.

74 **BCRDEX** **$/m^2**

See **ACRDEX**.

75 **CRDMC** **J/K**

Control rod drive mass times specific heat, for **ICREXP** = 1. Typical value: 5.6 x 10^4 J/K.

76 **CRDHA** **W/K**

Control rod drive surface area times heat- transfer coefficient, for **ICREXP** = 1. Typical value: 2300.

77 **UIVOL** **m^3**

Coolant volume in the upper internal structure region. Typical value: 25. Note: Locations 71 - 77 are only used if **ICREXP** > 0.

78 **RDEXPC** **$/K**

Coefficient in simple radial expansion feedback model.

79 **XMCXAC**

XMC/XAC in simple radial expansion feedback model. XMC: Distance from nozzle support point to core midplane. XAC: Distance from nozzle support point to above core load pad.

80-89 **SCRTAB** **$**

Scram reactivity table (See **ASCRAM**, **PSCRAM**, **GSCRAM**).

90-99 **SCRTME** **s**

Times for **SCRTAB**. Zero time corresponds to TSCRAM+DELSCR, where TSCRAM is the time for scram initiation determined by **ASCRAM**, **PSCRAM**, or **GSCRAM**.

100-179 **PRETB2 (K,IPW)**

Normalized power table for power type IPW (K = 1-20, IPW = 2-5). Note: Table for type 1 is in **PREATB**. Total reactor power reported in the output represents only channels assigned to power type 1 and does not include power from channels assigned to power types 2-5.

180-259 **PRETM2 (K,IPW)** **s**

Times for **PRETB2** table.

260-289 **BETADK (L,IPW)**

Decay heat precursor yield for group L in decay heat curve IPW. 1 ≤ L ≤ Min(**NDKGRP**, 6), 1 ≤ IPW ≤ **NPOWDK** Only decay heat curves with six or fewer groups may be defined by **BETADK**. For curves consisting of more terms, see **DKBET2**.

290-319 **DKLAM (L,IPW)** **1/s**

Decay heat decay constant for **BETADK**(L,IPW).

320-324 **BETAHT (IPW)**

Sum of decay heat precursor yields for user-supplied decay heat curve IPW. If **BETAHT**(IPW) > 0.0, precursor yields for curve IPW are renormalized to **BETAHT**(IPW). **BETAHT** applies to precursor yields defined by **BETADK** or **DKBET2**.

325-364 **POWLVL (K,IPR)**

Table of normalized total power for initializing decay power in decay heat region IPR. 1 ≤ K ≤ **NPDKST** ≤ 8 1 ≤ IPR ≤ Min(NDKREG, 5) NDKREG is the number of decay heat regions as determined internally by the code based on user-supplied decay heat input (see **DKFRAC** below). Only the first five regions are included in this table (see **PWLVL2**/**PWTIM2** below to specify the remaining regions). Zero values in this table will initialize decay heat based on zero total power. To calculate infinite, steady-state initialization for all regions, set **NPDKST** to zero.

365-404 **POWTIM (K,IPR)** **s**

Duration (in seconds) of initializing power level **POWLVL**(K,IPR).

405 **PUBYU**

Not currently used.

406 **HAUIS** **W/K**

Heat transfer coefficient * area for upper internal structure to hot pool heat transfer. Used for **ICREXP** > 0.

407 **XMCUI** **J/K**

Mass * specific heat of steel in the upper internal structure region. Used for **ICREXP** > 0.

LOCATIONS 408-415 USED ONLY IF |IRADEX| > 3

408 **SLLMAX** **m/m**

Maximum allowable slope of subassembly at grid plate with respect to vertical based on subassembly nozzle/grid plate clearances; default: 2.0×10^{-4}.

409 **PITCHG** m

Subassembly pitch at the grid at the reference temperature **TR** .

410 **PITCHA** m

Flat-to-flat dimension across the above core load pad at the reference temperature **TR**.

411 **PITCHT** m

Flat-to-flat dimension across the top load pad at the reference temperature **TR**.

412 **RDEXCF** $/m

Radial expansion coefficient for uniform core dilation.

413 **TLPRRC** m

Clearance between the top load pad and the restraint ring. Default: 2.54 x 10^{-3} m.

414 **BNDMM1** 1/m

Applied bending moment at the top of the core region, representing the flat-to-flat temperature difference at the outer edge of the active core. Default: 1.4 x 10^{-3}.

415 **BNDMM2** 1/m

Applied bending moment in the region above the core, representing the flat-to-flat temperature difference in this region for subassemblies at the outer edge of the active core. Default: 1.4 x 10^{-3}.

416 **TINSRT** s

Time interval over which **REAINS** dollars of reactivity is inserted. Default: 1.0.

417 **REAINS** $

Amount of reactivity to be inserted linearly during time interval **TINSRT**.

418 **TLIMIT** K

Control rod drive line temperature at which insertion of reactivity **REAINS** is to begin.

LOCATIONS 419-428 USED ONLY IF |IRADEX| > 3

419 **DFLTCS** m

Subassembly displacement at the above-core load pad at zero power resulting from creep and irradiation swelling history, positive outward, for subassemblies at the outer edge of active core.

420 **DFLTSS** **m**

Subassembly displacement at the top load pad at zero power resulting from creep and irradiation swelling history, positive outward, for the subassemblies at the outer edge of active core.

421 **ACLPRC** **m**

Clearance between the compacted above-core load pads and the restraint ring at the above-core load pad elevation, if any. If no above-core restraint ring, enter 0.

422 **FCDTR1**

Nominal steady-state above-core restraint ring temperature, expressed as a fraction of the average coolant temperature rise through the core.

423 **FCDTR2**

Nominal steady-state top restraint ring temperature, expressed as a fraction of the coolant temperature rise through the core.

424 **FCDTRF**

Nominal steady-state reflector load pad temperature, expressed as a fraction of the coolant temperature rise through the core.

425 **DRCOLL** **m**

Additional clearance between the subassembly and its load pad, known as a "floating collar".

426 **CRSAC** **m**

Additional clearance in the interior of the core, or the difference between the actual core radius and the ideal core radius. Default: 6.35×10^{-4}.

427 **RR1TC** **s**

Thermal response time constant for the above-core restraint ring.

428 **RR2TC** **s**

Thermal response time constant for the top restraint ring.

ADDITIONAL INPUT FOR THE DETAILED CONTROL ROD EXPANSION MODEL

REQUIRED FOR ICREXP = 4.

429 **RODID** m

Outside diameter of control rod driveline.

430 **RODOD** m

Outside diameter of control rod driveline.

431-433 **SHRDLN (K)** m

Length of section K of the control rod, section is deleted if zero.

434-436 **SHRDID (K)** m

Inside diameter of section of control rod shroud, no shroud assumed if zero.

437-439 **SHRDOD (K)** m

Outside diameter of section of control rod shroud, no shroud is assumed if zero.

440 **RHOCRD** kg/m^3

Density of control rod structure.

441 **HTCPCR** J/kg-K

Heat capacity of control rod structure.

442 **CONDCR** W/m-K

Thermal conductivity of control rod structure.

443 **VFCRD**

Structure volume fraction in driveline core, remainder is sodium.

444 **HFILM** W/m^2-K

Film coefficient on outer control rod shroud surface.

445 **FLSHRD** kg/s

Flowrate in shroud annulus.

446 **AREACR** m^2

Discharge area for segment representing control rod assembly(s).

447 **FLOEXP**

Exponent on flow in the CRD shroud friction pressure drop equation.

ADDITIONAL INPUT FOR THE DETAILED RADIAL CORE EXPANSION MODEL

REQUIRED FOR |IRADEX| > 3.

448 **ACLPEL** **m**

Elevation of the center of the above-core load pad with respect to the bottom of the fueled region, zone 'KZPIN', at the reference temperature **TR**.

449 **TLPEL** **m**

Elevation of the center of the top load pad with respect to the bottom of the fueled region, zone 'KZPIN', at the reference temperature **TR**.

450 **PTCHRA** **m**

Flat-to-flat dimension across the above-core load pad for subassemblies exterior to the last row of driver subassemblies at the reference temperature **TR**.

451 **PTCHRT** **m**

Flat-to-flat dimension across the top load pad for subassemblies exterior to the last row of driver subassemblies at the reference temperature **TR**.

452 **RCBARR** **m**

Core barrel radius at the reference temperature **TR**.

453 **FCDTCB**

Nominal steady-state core barrel temperature, expressed as a fraction of the coolant temperature rise through the core.

454 **CB2TC** **s**

Thermal response time constant for the core barrel.

ADDITIONAL INPUT FOR THE EBR-II REACTIVITY FEEDBACK MODEL, |IREACT| = 2

455 **YKNF** **Δk/(Δn/n)**

Fuel number-density coefficient of reactivity.

456 **YKHF** **Δk/(Δh/h)**

Fuel change in reactivity per fractional change in core fuel height.

457 **YKNNA** **Δk/(Δn/n)**

Sodium number-density coefficient of reactivity.

458 **YKNSS** **Δk/(Δn/n)**

Steel number-density coefficient of reactivity.

459 **YRCUR** **Δk/K**

Upper-reflector coefficient of reactivity.

460 **YLCLR** **Δk/K**

Lower-reflector coefficient of reactivity.

461 **YRCRR** **Δk/K**

Radial-reflector coefficient of reactivity.

462 **YRCCR** **Δk/K**

Control-rod-flow coefficient of reactivity.

463 **YRCGP** **Δk/K**

Grid-plate coefficient of reactivity.

464 **YRCDOP** **Δk**

Non-linear Doppler-effect of reactivity.

465 **YDELT0** **K**

Nominal core delta T at full reactor power.

466 **YABOW**

Coefficient of non-linear core bowing effect. P = B Delta T/Delta T Deg. + A. (See **YBBOW**).

467 **YBBOW**

Coefficient of non-linear core bowing effect. P = B Delta T/Delta T Deg. + A. (See **YABOW**).

468 **FCR**

Control-rod feedback parameter: 0 ≤ FCR ≤ 1.

469 **YTCUT**

The normalized core temperature rise below which the bowing feedback is 0. For instance, if **YTCUT**=0.5, then for core temperature increases that are less than 1/2 of the nominal core temperature rise (103.5 K) reactor feedback due to bowing is 0$.

EXTERNAL SOURCE SPECIFICATION FOR POINT AND SPATIAL KINETICS APPLICATIONS

470 **RHOZRO** $

Initial subcritical reactivity for point kinetics external source. =0, No external source. <0, Initial external source will be set to give a steady initial steady state with the reactivity equal to **RHOZRO**.

471-490 **EXSOTB (L)**

Relative point and spatial kinetics external source values at times given in **EXSOTM**.

491-510 **EXSOTM (L)** s

Times for point and spatial kinetics external source values given in **EXSOTB**. See also **NEXSO** and **RHOZRO**. The time dependence for the point kinetics external source specified by **RHOZRO** will be given by the pairs of values entered in **EXSOTB** and **EXSOTM**. For RHOZRO < 0 and NEXSO = 0, a constant external source will be used. The **EXSOTB** values will be normalized to unity at t = 0. The time dependence for the spatial kinetics external source specified on the FIXSRC FILE will be given by the pairs of values entered in **EXSOTB** and **EXSOTM**. For NEXSO = 0, a constant external source will be used. The **EXSOTB** values will be normalized to unity at t = 0.

511-630 **DKBET2 (L,IPW)**

Decay heat precursor yield for group L in decay heat curve IPW. 1 ≤ L ≤ **NDKGRP** ≤ 24 1 ≤ IPW ≤ **NPOWDK** ≤ 5 Decay heat data for curve IPW may be present in either **BETADK**/**DKLAM** or **DKBET2**/**DKLAM2**. If data is present in both locations, values in **DKBET2**/**DKLAM2** will be used.

631-750 **DKLAM2 (L,IPW)** 1/s

Decay heat decay constant for **DKBET2**(L,IPW).

751-800 **DKFRAC (IPR,IPW)**

Fraction of *user-supplied* decay heat curve IPW to be used in decay heat region IPR. 1 ≤ IPR ≤ NDKREG ≤ 10 1 ≤ IPW ≤ **NPOWDK** ≤ 5 NDKREG is determined internally by the code based on the input of **DKFRAC** and **DKANSI** (below). A maximum of 10 regions can be defined. By default, **DKFRAC** is a **NPOWDK** by **NPOWDK** identity matrix, providing compatibility with old input files. (Regions and curves have the same meaning in this case).

801-880 **DKANSI (IPR,N)**

Fraction of *built-in* ANS standard decay curve N to be used in decay heat region IPR.

N = 1: U-235 thermal fission
N = 2: Pu-239 thermal fission
N = 3: U-238 fast fission
N = 4: Pu-241 thermal fission
5 ≤ N ≤ 8 is reserved for future standard curves.

881-920 **PWLVL2 (K,IPR-5)**

Table of normalized total power for initializing decay power in decay heat region IPR. 1 ≤ K ≤ **NPDKST** ≤ 8 6 ≤ IPR ≤ NDKREG ≤ 10 **PWLVL2** is a continuation of table **POWLVL**.

921-960 **PWTIM2 (K,IPR-5)** **s**

Duration of initializing power level **PWLVL2**(K,IPR-5). **PWTIM2** is a continuation of table **POWTIM**.

961-968 **QETOT (N)** **MEV/fission**

Total recoverable energy per fission for the fissionable isotope associated with built-in standard decay curve N. See **DKANSI**. Default is 200 MeV/fission.

N = 1: U-235 thermal fission
N = 2: Pu-239 thermal fission
N = 3: U-238 fast fission
N = 4: Pu-241 thermal fission
5 ≤ N ≤ 8 is reserved for future standard curves.

969-1000 **DUMPNA**

Not currently used.

2.7.7 Block 13 — PMATCM — Fuel and Cladding Properties

Note: Suggested values refer to oxide fuel.

1 **COEFDS (1)** **kg/m^3**

Solid fuel theoretical density at the reference temperature. Suggested value: 11.08E+3. Theoretical fuel density = **COEFDS**(1) / [1.0 + (**COEFDS**(2) + **COEFDS**(3) * (TK-273.15)) * (TK-273.15)].

2 **COEFDS (2)** **1/K**

First-order solid fuel volumetric thermal expansion coefficient, ALPHA1. Suggested value: 2.04E-5.

3 **COEFDS (3)** **1/K^2**

Second-order solid fuel volumetric thermal expansion coefficient, BETA1 Suggested value: 8.70E-9. Used only if **IRHOK** > 0.

4-10 **COEFK (1-7)**

Fuel thermal conductivity coefficients. Fuel thermal conductivity = ((**COEFK**(1) - FDEN) * FDEN - 1.0) * ((**COEFK**(2) + **COEFK**(3) * TK)$^{-1}$ + **COEFK**(4) * TK^3) if FDEN ≤ 0.95, and thermal conductivity = (3.0 * FDEN -1.0) * ((**COEFK**(5) + **COEFK**(6) * TK)$^{-1}$ + **COEFK**(7) * TK^3) if FDEN > 0.95, where FDEN is fractional fuel density. Suggested values: 2.1, 2.88E-3, 2.52E-5, 5.83E-10, 5.75E-2, 5.03E-4, 2.91E-11. Used only if **IRHOK** = 2.

11-70 **EXKTB (L,ICLAD)** **W/m-K**

Table of clad thermal conductivity values for clad *ICLAD*, 1 ≤ *L* ≤ 20. See **EXKTM** for corresponding temperature values.

If only **EXKTB**(1) is entered, property is temperature-independent.

(1 ≤ *ICLAD* ≤ **ICLAD1**)

71-90 **EXKTM (L)** **K**

Temperature values for **EXKTB** table, 1 ≤ *L* ≤ 20.

91-250 **RHOTAB (L,IFUEL)** **kg/m^3**

Table of theoretical fuel density for fuel *IFUEL*, 1 ≤ *L* ≤ 20. See **RHOTEM** for corresponding temperature values.

Used for **IMETAL** = 0. See **RHOZN**for **IMETAL** > 0.

251-410 **RHOTEM (L,IFUEL)** **K**

Temperature values for **RHOTAB** table, 1 ≤ *L* ≤ 20, for fuel *IFUEL*.

Note: If fuel solidus and liquidus temperatures are defined for fuel melting (see **TFSOL** and **TFLIQ**), then two adjacent entries for these temperatures should be provided in this table.

411-418 **TMF (IFUEL)** **K**

Not to be input. It is set to fuel solidus temperature in the code. See **TFSOL**.

419 **TR** **K**

Reference design point temperature. Temperature at which pin dimensions are measured. Suggested value: 300.

420-579 **XKTAB (L,IFUEL)** **W/m-K**

Table of fuel thermal conductivity values for fuel *IFUEL*, $1 \le L \le 20$. See **XKTEM** for corresponding temperature values.

If only **XKTAB**(1) is entered, property is temperature-independent.

($1 \le IFUEL \le$ **IFUEL1**)

580-599 **XKTEM (L)** **K**

Temperature values for **XKTAB** table, $1 \le L \le 20$.

600 **FGMM** **g/mol**

Molecular weight of fission-gas atom. Suggested value: 131.

601 **GATPF**

Gas atoms generated per fission. Suggested value: 0.246.

602 **ENPF** **MeV**

Energy per fission. Suggested value: 197.

603 **RLEQ**

Fraction of the as-fabricated porosity which defines the equiaxed / columnar fuel boundary. Suggested value: 0.6. PRSTY2 < **RLEQ** * **PRSTY** - columnar fuel PRSTY2 > **RLEQ** * **PRSTY** - equiaxed fuel

604 **RUEQ**

Grain size ratio at the equiaxed / as-fabricated boundary. Suggested value: 1.2. DGR2 < **RUEQ** * **DGO** - as-fabricated DGR2 ≥ **RUEQ** * **DGO** - equiaxed

605 **PRSMIN**

Minimum attainable porosity during restructuring. Suggested value: 0.02.

606-765 **CPFTAB (L,IFUEL)** **J/kg-K**

Table of fuel specific heat values for fuel *IFUEL*, $1 \le L \le 20$. See **CPFTEM** for corresponding temperature values.

If only **CPFTAB**(1) is entered, property is temperature-independent.

($1 \le IFUEL \le$ **IFUEL1**)

766-785 **CPFTEM (L)** **K**

Temperature values for **CPFTAB** table, $1 \le L \le 20$.

786-793 **TFSOL (IFUEL)** **K**

Fuel solidus temperature.

794-801 **TFLIQ (IFUEL)** **K**

Fuel liquidus temperature.

802-809 **UFMELT (IFUEL)** **J/kg**

Fuel heat of fusion.

See also **HFMELT**.

810-812 **TESOL (ICLAD)** **K**

Cladding solidus temperature.

813-815 **TELIQ (ICLAD)** **K**

Cladding liquidus temperature.

816-818 **UEMELT (ICLAD)** **J/kg**

Cladding heat of fusion.

819-878 **CPCTAB (L,ICLAD)** **J/kg-K**

Table of clad specific heat values for clad *ICLAD*, $1 \le L \le 20$. See **CPCTEM** for corresponding temperature values.

If only **CPCTAB**(1) is entered, property is temperature-independent.

($1 \le ICLAD \le$ **ICLAD1**)

879-898 **CPCTEM (L)** **K**

Temperature values for **CPCTAB** table, $1 \le L \le 20$.

899-901 **TME (ICLAD)** **K**

Not to be input. Set to cladding solidus temperature in the code. See **TESOL**.

902-909 **UMELT (IFUEL)**

Not currently used.

910-969 **YLDTAB (L,ICLAD)** **Pa**

Table of clad yield values for clad *ICLAD*, $1 \le L \le 20$. See **YLDTEM** for corresponding temperature values.

If only **YLDTAB**(1) is entered, property is temperature-independent.

(1 ≤ *ICLAD* ≤ **ICLAD1**)

970-989 **YLDTEM (L)** **K**

Temperatures values for **YLDTAB** table.

990-1049 **CROETB (L,ICLAD)** **J/m^3-K**

Specific heat * density for cladding in table location L, 1 ≤ *L* ≤ 20. See **CROETM** for corresponding temperature values.

If only **CROETB**(1) is entered, property is temperature-independent.

1050-1069 **CROETM (L)** **K**

Temperature values for **CROETB** table.

1070-1072 **CE (ICLAD)** **J/kg-K**

Cladding specific heat at solidus temperature for CLAP cladding motion module. 1 ≤ ICLAD ≤ 3. Suggested value: 690.

1073-1080 **PRSTY (IFUEL)**

As-fabricated porosity for each fuel type. IFUEL ≤ 8.

1081 **AC**

Not currently used.

1082 **QSWL**

Not currently used.

1083 **APORE** **m^2-K^1.5/s**

Pre-exponential factor in pore velocity.

VPORE = **APORE** * (dT/dr) * exp(-**QPORE**/(**RGASSI** * T)) / (T ** **ABC**)

Suggested value: 20.704.

1084 **QPORE** **J/gm-mole**

Pore migration activation energy Suggested value: 4.5281E05.

1085 **ABC**

Exponent of temperature factor in pore velocity. Suggested value: 1.5.

1086 **RGASSI** J/gm-mole-K

Ideal gas constant. Suggested value: 8.31434.

1087 **GAMMA** J/m^2

Fuel surface tension. Suggested value: 0.45. See **SUFU**.

1088 **APG** Pa

Pre-exponential factor in bubble radius parameterization. Suggested value: 5.0E04. | RB = (2 * **GAMMA**/**APG**) * exp (-**QPG** / (**RGASSI** * TEMP))

1089 **QPG** J/mole

Temperature dependence of bubble radius. Suggested value: 5.65065E04.

1090 **GK** m^3/s

Pre-exponential factor in unlimited grain growth rate. DG ** **NGRAIN** = **DGO** ** **NGRAIN** + **GK** * TIME * exp (- **QV** / (**RGASSI** * TEMP)) Suggested value: 1.717E10.

1091 **QV** J/gm-mole

Activation energy of unlimited grain growth rate. Suggested value: 3.87E05.

1092 **GK1** m^2/s

Pre-exponential factor in limited grain growth rate. (Ainscough et al. model). Suggested value: 1.45556E-08.

1093 **QV1** J/mole-K

Temperature dependence in limited grain growth rate. Suggested value: 2.67E+5.

1094 **GRAINK** m

Pre-exponential factor in maximum grain size. DMAX = **GRAINK** * exp(-**GRAINQ** / (**RGASSI*** TEMP))

1095 **GRAINQ** J/gm-mole

Temperature dependence in maximum grain size. Suggested value: 6.3375E+04.

1096 **CVXE** J/kg-K

Xenon specific heat at constant volume. Suggested value: 94.69.

1097 **CVHE** J/kg-K

Helium specific heat at constant volume. Suggested value: 3.13E+03.

1098 **ROFF** m

Surface roughness of fuel. Suggested value: 3.3E-06.

1099 **ROFC** m

Surface roughness of inner cladding. Suggested value: 1.78E-06.

1100 **AZEROX**

Calibration factor in solid-solid conductance. Not currently used.

1101 **GAMGS**

Cp/Cv parameter in jump distance calculation. Suggested value: 1.66.

1102 **HARDNS** Pa

Meyer's hardness of the softer contacting surface in solid-solid gap conductance. This is set to 3 * yield stress within the code.

1103 **ET**

Exponent of the pressure dependence in solid-solid conductance. Suggested value: 1.

1104 **ACCHE**

Accommodation coefficient of helium. Suggested value: 0.15.

1105 **ACCXE**

Accommodation coefficient of xenon. Suggested value: 0.805.

1106 **CZERO**

Calibration constant for surface roughness in gap conductance calculation. Suggested value: 1.98.

1107 **STEBOL** W/m^2-K^4

Stefan-Boltzmann constant. Suggested value: 5.67E-08.

1108 **EMSF**

Fuel emissivity. Suggested value: 0.9.

1109 EMSC

Cladding emissivity. Suggested value: 0.8.

1110 QA1 K

Parameter in Weisman fission-gas release model. Probability of escaping directly without being trapped, KPRIME = exp(-**QA1**/ TEMP - **QA2** - **QA3** * DEN) where DEN = percent of fuel theoretical density. Suggested value: 6.92E+3.

1111 QA2

Parameter in Weisman fission-gas release model. Suggested value: 33.95.

1112 QA3

Parameter in Weisman fission-gas release model. Suggested value: 0.338.

1113 QA4 K

Parameter in Weisman fission-gas release model. Probability of trapped fission-gas atom getting released from trap and then escaping per second, K = exp(-**QA4**/ TEMP - **QA5**). K is in units of 1/s. Suggested value: 1.48E+04.

1114 QA5

Parameter in Weisman fission-gas release model. Suggested value: 9.575.

1115 ALFSS 1/s

Pre-exponential factor in isotropic fission- gas release model. Fractional release rate= **ALFSS** * exp(-**BETSS**/TEMP). Suggested value: 2.0E-04.

1116 BETSS K

Temperature dependence in isotropic fission-gas release model. Suggested value: 1.1E+04.

1117 CNU

Cladding Poisson ratio. Suggested value: 0.3.

1118 FNU

Fuel Poisson ratio. Suggested value: 0.3265.

1119 AM

Stress exponent in the fuel creep law for stress relaxation calculation. Strain rate = AONE * stress ** **AM** * exp(-**QLAX**/(**RGASSI** * Temp)). "AONE" is evaluated in the code depending on fractional fuel density and fuel grain size. Suggested value: 1.0.

1120 **QLAX** J/mole

Activation energy for diffusion creep. Suggested value: 3.77E+05.

1121 **QLAX2**

Not currently used.

1122 **DDX**

Not currently used.

1123 **DDX2**

Not currently used.

1124 **RGAS** Pa-m^3/kg-K

Gas constant per kilogram for fission-gas (as in PV = MRT). Suggested value: 65 for high burnups.

PLUTO2 AND LEVITATE Input (1125-1199, 1206-1217, 1229-1257)

1125 **CINAF0**

Fraction of the coolant flow area occupied by the liquid sodium film left behind by moving coolant slugs in PLUTO2. This is also the maximum value in the voided channel. **CINAF0** must be < **CIVOID**. See **WFS00** and **WF0** to input consistent values. Suggested value: 0.15.

1126 **CIBBIN**

If the liquid fuel volume fraction in the channel is greater than **CIBBIN**, the fuel flow regime is treated as a bubbly flow regime. See **VFNALQ** and **CIANIN**.

1127 **CIREFU**

Reynolds number for annular or bubbly fuel flow above which the friction factor is assumed to be constant and equal to **CIFRFU**.

1128 **CIFRFU**

Moody friction factor for turbulent bubbly fuel flow (See **CIREFU**).

1129 **CIFUMO**

Fraction of the axial momentum of the fuel flow in the pin which is retained by the fuel which is ejected into the coolant channel.

1130 **CIVOID**

If the sodium void fraction in the coolant channel is less than **CIVOID** the heat-transfer and friction between sodium and cladding are based on single- phase correlations for the homogeneous mixture

of the two phases of sodium. **CIVOID** must be > **CINAF0** and < **CIA4**. For sodium void fractions greater than **CIVOID** see **CIA4** and **HCFFMI**. See Table 14.4.1.

1131 CIA1

Constant in the fuel particle-to-sodium heat-transfer coefficient. H = **CIA1***(conductivity of fuel)/(particle radius) * (1-Na void fraction) ** **CIA2**. (See Eq. (14.4-112) and Eq. (14.4-116)).

1132 CIA2

(See **CIA1**).

1133 CIA3

Constant in the molten fuel-to-cladding and molten fuel-to-solid fuel heat-transfer coefficient. Constant for oxide fuel but strongly dependent on Reynolds number for metal fuel. (See Eq. (14.2-36), Eq. (14.4-144), and Eq. (14.4-152)).

1134 CIA4

If the sodium void fraction is greater than **CIA4**, the heat-transfer coefficient between two-phase sodium and cladding, moving fuel and fuel crust is based on an interpolation. **CIA4** must be > **CIVOID**. (See Eq. (14.4-103), Eq. (14.4-125) and Eq. (14.4-130) in the documentation).

1135 CIA5

Constant in the fuel particle-to-sodium / fission-gas drag which controls the dependence on the void fraction. (See Eq. (14.4-195) in the documentation). Suggested value: -1.7.

1136 CIA6

Constant in the bubbly fuel flow drag calculation (See Eq. (14.4-202)). Suggested value: 0.4272.

1137 CIFN

Not currently used.

1138 FNFUAN

Not currently used.

1139 CPFU J/kg-K

Average specific heat of moving liquid or solid fuel.

1140 CDFU J/m-s-K

Average thermal conductivity of moving liquid or solid fuel.

1141 **CMNL** **Pa^-1**

Average compressibility of liquid sodium at roughly the sodium temperature at the time of pin failure in the vicinity of failure.

1142 **CDNL** **J/m-s-K**

Average thermal conductivity of liquid sodium at roughly the sodium temperature at the time of pin failure in the vicinity of failure.

1143 **CIETFU**

Effectiveness of fuel particles to entrain or tear off a liquid sodium film. **CIETFU** = 1.0 means that the fuel acts as a gas with the gas density equal to the fuel smear density. For **CIETFU** = 0.1 only one tenth of the fuel density is used.

1144 **CDVG** **J/m-s-K**

Average thermal conductivity of the sodium vapor / fission-gas mixture at temperatures roughly 200-300 K higher than the sodium temperature at the time of pin failure in the vicinity of failure. Used only in Dittus-Boetter correlation for Nusselt numbers. At present, use sodium vapor value.

1145 **VIFI** **kg/m-s**

Average viscosity of fission-gas.

1146 **CFNACN** **J/m^2-s-K**

Sodium condensation coefficient.

1147 **CFNAEV** **J/m^2-s-K**

Sodium evaporation coefficient. Should be larger than the above condensation coefficient.

1148 **FIFNGB**

Fraction of the fission-gas entering the cavity with the melting-in fuel which is on grain boundaries. This fraction of the fission-gas becomes immediately available, whereas the remainder becomes available only after a coalescence time. Suggested value: 0.10. (See **CIRTFS**)

1149 **VINL** **kg/m-s**

Average viscosity of liquid sodium at roughly the sodium temperature at the time of pin failure in the vicinity of failure.

1150 **VIVG** **kg/m-s**

Average viscosity of the fission-gas/ sodium-vapor mixture. (For temperature range and other comments, see **CDVG**).

1151 **EGFUSO** J/kg

Internal energy of fuel at the solidus point.

1152 **DZPLIN** m

Minimum length of the edge cells in the interaction region of PLUTO-2 and LEVITATE. Values between 0.005 and 0.05 meter are allowed. **DZPLIN** has to be smaller than the shortest mesh cell in all channels. Recommended and default value: 0.02.

1153 **CFCOFV**

Condensation heat-transfer coefficient for fuel vapor. :units: J/m^2-s-K

1154 **CFFURH**

Not currently used.

1155 **C1VIPR**

A dimensionless constant in the artificial viscous pressure calculation inside the pin. (See Eq. (14.2-46))

1156 **C2VIPR**

A dimensionless constant in the artificial viscous pressure calculation inside the pin. (See Eq. (14.2-45))

1157 **SUFU** J/m^2

Molten fuel surface tension. (See **GAMMA**)

1158 **RAFPLA** m

Radius of initial fuel particles.

1159 **RAFPSM** m

Radius of fuel particles after fragmentation, after **TIFP** s from initial injection.

1160 **VFNALQ**

Liquid sodium volume fraction in the coolant channels below which a continuous (annular or bubbly) fuel flow regime can be initiated (See **CIBBIN**, **CIANIN**, and **EGMN** used in determining fuel flow regimes).

1161 **EGBBLY** J/kg

If the fuel flow regime is annular or bubbly, fuel freezing may be initiated when the fuel internal energy drops below EGBBLY. Its value should be above the solidus energy.

1162 VIFULQ kg/m-s

Viscosity of the fuel above the fuel liquidus.

1163 VFNARE

Liquid sodium volume fraction above which a continuous fuel flow becomes a particulate flow again. Suggested value: >0.5. (See Figure 14.4.4)

1164 DTPLIN s

Initial and minimum PLUTO2 and LEVITATE time step. Suggested value: 2.E-5. Minimum value: 1.E-6. Maximum value: 2.E-4.

1165 AXMX m^2

Reference area for PLUTO2 and LEVITATE. This area times 1 meter is the volume to which all volume fractions in PLUTO2 or LEVITATE are referenced. It makes the volume fractions more meaningful if this area is equal to the area encompassing everything inside the outer perimeter of the subassembly wall (not per fuel-pin but per fuel subassembly).

1166 EPCH

Not currently used.

1167 TIPLMX s

Time after PLUTO2 initiation when full PLUTO2 calculations are switched off and only the PLUTO2 energy equations are solved for all components which are then assumed to remain stagnant. Necessary for transient overpower calculations in which the lead channel fails many seconds before any other channel.

1168 DTPLP s

Full PLUTO2 or LEVITATE printout every **DTPLP** seconds.

1169 FNMELT

Molten fuel is added to the cavity when it has gone through a fraction **FNMELT** of the heat of fusion. Suggested value: 0.9 (for TREAT experiment analysis lower values may be necessary).

1170 CIRTFS 1/s

Determines how fast dissolved fission-gas in the pin cavity coalesces and becomes free gas: Mass of fission-gas coalescing per unit time = **CIRTFS** * current mass of dissolved fission-gas in this node. Suggested value: 16.667.

1171 CISP

Not currently used.

1172 CIFUFZ

Controls the mode of fuel freezing in PLUTO2.

= 0, Conduction type freezing.
= 1, Bulk type freezing.

Allowed range between 0.0 to 1.0. Suggested value: 1.

1173 TIFP s

Time delay for fragmentation of larger particles into smaller ones (relative to initial fuel injection time, see **IPSIZE**).

1174 CIANIN

Channel fuel (moving fuel + frozen fuel) volume fraction above which the whole perimeter of the channel is wetted by molten fuel in annular flow.

1175 TEFAIL K

Cladding temperature of a node above which the cladding failure propagates to this node if the pin pressure is greater than the channel pressure + **PRFAIL** and also the areal fuel melt fraction greater than **FNARME**. Both cladding nodes must exceed this temperature. Relevant only if axial pin failure propagation is determined by input, if **KFAILP** = 1. Suggested value: Steel solidus temperature **TESOL**.

1176 FNARME

Relevant for both axial pin failure propagation options (**KFAILP** = 0 or 1). FARME is the minimum areal fuel melt fraction of a node above which a cladding failure can propagate into this node if additional conditions depending upon the value of **KFAILP** are satisfied.

1177 PRFAIL Pa

Relevant only if **KFAILP** = 1. (See **TEFAIL**).

1178 EGMN J/kg

The continuous fuel flow regimes cannot be initiated below this fuel internal energy. Its value should be above the solidus energy.

1179 HCFFMI J/m^2-s-K

Convective heat-transfer coefficient from the surface of a frozen fuel crust to two-phase sodium/fission-gas mixture (liquid sodium in the form of dispersed drops) with a void fraction > **CIVOID**.

1180 **HCFUBB** J/m^2-s-K

Convective heat-transfer coefficient between the interior of the molten fuel and two-phase sodium/fission gas mixture bubble surfaces in the bubbly flow regime.

1181 **FNHTFU**

Fraction of the convective heat-transfer coefficient between liquid fuel and cladding which remains effective when the moving fuel consists of solid chunks.

1182 **XPL5**

For future use in PLUTO2.

1183 **XPL6**

For future use in PLUTO2.

1184 **TECLMN** K

Maximum outer cladding node temperature above which freezing fuel cannot stick to the cladding (the same input value limits the freezing on the inner structure node). (this is used in PLUTO2 only).

1185 **TECLRL** K

Temperature of the middle cladding node above which plated-out fuel is released. (this is used in PLUTO2 only).

1186 **CIHCFU**

Dimensionless coefficient in the Deissler heat-transfer correlation for inpin fuel motion. Nu = **CIHCFU***Pr*$Re^{0.8}$

1187 **HCCLMI** J/m^2-s-K

Convective heat-transfer coefficient from hotter cladding to two-phase sodium/ fission-gas mixture (liquid sodium in the form of dispersed drops) with a void fraction > **CIVOID**.

1188 **CMFU** Pa^-1

Average adiabatic compressibility of liquid fuel.

1189 **XPL7**

For future use in PLUTO2.

1190 **XLP8**

For future use in PLUTO2.

1191 **XLP9**

For future use in PLUTO2.

1192 **XPL10**

For future use in PLUTO2.

1193 **XPL11**

For future use in PLUTO2.

1194 **XPL12**

For future use in PLUTO2.

1195 **CDCL** **J/m-s-K**

Average conductivity of the solid cladding.

1196 **CPCL** **J/kg-K**

Average specific heat of the solid cladding.

1197 **CPCLRH** **J/m^3-K**

Average specific heat times density of the solid cladding.

1198 **RHSLBT** **kg/m^3**

Average physical density of the lower liquid sodium slug.

1199 **RHSLTP** **kg/m^3**

Average physical density of the upper liquid sodium slug.

1200 **COEFDL (1)**

Not currently used.

1201 **COEFDL (2)** **1/K**

Molten fuel volumetric thermal expansion coefficient, ALPHA2. (DEFORM uses with **COEFDS**(1)) Suggested value: 9.3E-5. Rho = **COEFDS**(1)/(1+**COEFDL**(2) *(T-273.15))

1202 **QSTAR**

Not currently used.

1203 ABCPU

Not currently used.

1204 QPU

Not currently used.

1205 DPUO

Not currently used.

1206 RHSSLQ kg/m^3

Density of steel in the fuel crust. (See **CPCL**, **CPCLRH**, and **RHSSSO**). Suggested value: 6000.

1207 CIBBDI

Liquid fuel volume fraction above which the transition from annular to bubbly fuel flow regime occurs in disrupted regions (i.e., regions having no pin geometry). Suggested value: 0.2.

1208 CIANDI

Liquid fuel volume fraction above which the transition from partial perimeter annular to full perimeter fuel flow regime occurs in disrupted regions. Suggested value: 0.1.

1209 CIVIMT

The fuel viscosity at solidus is set equal to **CIVIMT** times the value at liquidus (See **VIFULQ**). The fuel viscosity between solidus and liquidus is evaluated by interpolation between these two values. Suggested value: 200.

1210 EGSESO J/kg

Solidus energy of steel. Suggested value: 8.18E+5.

1211 EGSELQ J/kg

Liquidus energy of steel. Suggested value: 1.076E+6.

1212 CPSE J/kg-K

Specific heat of molten steel. (See **CPCTAB**, **CE**, **CPCL**, and **CPCLRH**). Suggested value: 768.

1213 FRMRSE

The fraction of latent heat of fusion to be satisfied for steel to be considered a moving fluid in LEVITATE. Suggested value: 0.5.

1214 FNSROS

Initial fraction of structure in the inner node (facing the channel). (See **DSTIZ** and **DSTOZ** for consistent input). Suggested value: 0.1 to 0.9.

1215 RHSSSO kg/m^3

Density of solid steel. (See **CPCL**, **CPCLRH** and **RHSSLQ**). Suggested value: 6.95E+3.

1216 RGFV J/kg-K

Gas constant for the fuel vapor. Suggested value: 31.

1217 RGSV J/kg-K

Gas constant for the steel vapor. Suggested value: 145.

1218 TMIDFG

Not currently used.

1219 FGSPRD

Not currently used.

1220 FGPORX

Not currently used.

1221 WST

Not currently used.

1222 HECOND

Not currently used.

1223 FGCOND

Not currently used.

1224 AKCOND

Not currently used.

1225 HEMM

Molecular mass of helium atom. Suggested value: 4.

1226 **EPSSFP** **Per^a/o burn-up**

Volume swelling fraction due to solid fission products. Suggested value: 3E-03.

1227 **HEMASX**

Not currently used.

1228 **ZSWFAC**

Not currently used.

1229 **FNDISR**

Ratio of molten cavity radius to fuel pellet radius needed for disrupting pins in LEVITATE. Pins can also disrupt if the outermost fuel node is above the solidus. (**DTDISR** must also be met).

1230 **DTDISR** **K**

Fuel-pin disruption in LEVITATE is allowed if the cladding middle node temperature is greater than cladding solidus temperature minus **DTDISR**. (**FNDISR** must also be met). Not currently used.

1231 **SRFMLE**

If **SRFMLE** = 0.0, liquid sodium film on structure is thrown away in LEVITATE. If **SRFMLE** > 1.E-10, the sodium in the structure film is conserved. LEVITATE mixes the sodium film with the sodium vapor in the channel. This can lead to an excessive pressure event in the channel due to fuel/sodium interactions.

1232 **XLE4**

Not currently used.

1233 **XLE5**

Not currently used.

1234 **XLE6**

Not currently used.

1235 **XLE7**

Not currently used.

1236 **XLE8**

Not currently used.

1237 **XLE9**

Not currently used.

1238 **XLE10**

Not currently used.

1239 **XLEPT1**

Not currently used.

1240 **XLEPT2**

Not currently used.

1241 **XLEPT3**

Not currently used.

1242 **XLEPT4**

Not currently used.

1243 **XLEPT5**

Not currently used.

1244 **XLEPT6**

Not currently used.

1245 **XLEPT7**

Not currently used.

1246 **XLEPT8**

Not currently used.

1247 **XLEPT9**

Not currently used.

1248 **PLUT1**

Not currently used.

1249 **PLUT2**

Not currently used.

1250 **PLUT3**

Not currently used.

1251 **PLUT4**

Not currently used.

1252 **PLUT5**

Not currently used.

1253 **PLUT6**

Not currently used.

1254 **PLUT7**

Not currently used.

1255 **PLUT8**

Not currently used.

1256 **PLUT9**

Not currently used.

1257 **PLUT10**

Not currently used.

1258 **FAXIAL**

Fraction of calculated axial expansion to be actually used by DEFORM. Suggested value: 1. (See **EXPCOF**).

1259 **FCLDWK**

Initial cold-work strain of the cladding. (Required if **IYLD** = 1).

1260 **FMELTD**

Not currently used.

1261 FSTRAN

Not currently used.

1262 FTMPCH

Fraction of fuel or blanket melt temperature (**TMF**(IFUEL)) at which crack healing is assumed. (Required if IHEALC = 2).

1263 EXPCOF

Fraction of the axial expansion reactivity calculated by DEFORM to be used for feedback effects. (See **FAXIAL**). 0.0 ≤ **EXPCOF** ≤ 1.0. Suggested value: 1.0.

1264 PRTSTR s

Time for the first transient printout if **IPROPT** = 1.

1265 PRTDEL s

Time interval between printouts if **IPROPT** = 1.

1266 FIRLIM

Failure fraction at which the main time step is cut to DTFALL. For each fuel pin failure option a time-dependent fraction between 0.0 and 1.0 called failure fraction is defined for each fuel axial segment which indicates how close to pin failure the axial segment is at the current time. Failure fraction is 0.0 at the beginning of the transient, and equals 1.0 at the time of pin failure. (See **MFAIL**).

1267 SECLIM

Failure fraction at which the main time step is cut to **DTFAL2**.

1268 THRLIM

Failure fraction at which the main time step is cut to **DTFAL3**.

1269 DTFAL1 s

Main time step when **FIRLIM** ≤ failure fraction < **SECLIM**.

1270 DTFAL2 s

Main time step when **SECLIM** ≤ failure fraction < **THRLIM**.

1271 DTFAL3 s

Main step when **THRLIM** ≤ failure fraction < 1.0.

1272 **AKD**

Not currently used.

1273 **CKD**

Not currently used.

1274 **QKD**

Not currently used.

1275 **FGFI**

Mole fraction of fission gas in the initial fill-gas. (Assumed same molecular weight as **FGMM**).

1276 **CIPINJ**

Controls the ejection of molten fuel/ fission gas from the pin cavity when the mechanistic ejection model is not used. See INRAEJ. Suggested value: 2.5E4.

1277 **DTPNIN** s

Initial and minimum PINACLE time step. Suggested value: 2.E-5.

1278 **TIPNMX** s

PINACLE maximum time. Not currently used.

1279 **DTPNP** s

Full PINACLE printout every **DTPNP** seconds.

1280 **ASRALU**

Aspect ratio of cylindrical chunks = 2R/L. Recommended value = 1.

1281 **UN1281**

Not currently used.

1282 **UN1282**

Not currently used.

1283 **UN1283**

Not currently used.

1284 RALUDI

Radius of chunks generated by pin disruption.

1285 RALUFZ

Radius of chunks generated by freezing and crust break-up. If set to zero the code provides default values based on local geometry. Recommended value: 0.

1286 CINAPN

= 0, All the sodium vapor pressure is added to the local pressure.
= 1, Only the excess of the sodium vapor pressure over the fission gas and fuel vapor pressure is added to the local pressure.

Note: This variable is relevant only if **INAPN** is equal to 1.

1287 CPCM

Chemical equilibrium coefficient of Pu at center-to-middle zone interface used in computing zone formation in U-Pu-Zr alloy fuel. Suggested value: 1.177.

1288 CPMO

Chemical equilibrium coefficient of Pu at center-to-outer zone interface used in computing zone formation in U-Pu-Zr alloy fuel. Suggested value: 1.0723.

1289 CZCM

Chemical equilibrium coefficient of Zr at center-to-middle zone interface. Suggested value: 9.79.

1290 CZMO

Chemical equilibrium coefficient of Zr at center-to-outer zone interface. Suggested value: 7.84.

1291 CUCM

Chemical equilibrium coefficient of U at center-to-middle zone interface. Suggested value: 0.448.

1292 CUMO

Chemical equilibrium coefficient of U at center-to-outer zone interface. Suggested value: 0.5866.

1293 EPSMS kg

Criterion for zonal fuel mass convergence in computing zone formation in U-Pu-Zr alloy fuel. Suggested value: 1.0E-6

1294 EPSCOM

Criterion for the convergence of zonal weight fractions of Pu and Zr. Suggested value: 1.0E-3

1295 CIPNTP

Controls the calculation of the fuel pin top boundary temperature, which is important in triggering the axial in-pin fuel relocation.

= 0, Boundary temperature is equal to the temperature of the material in the node above the active fuel.

= 1, The boundary temperature is the same as the temperature of the central top node of the active fuel.

= 2, This option is available for metal fuel pins only. The in-pin fuel motion is triggered by a mechanistic model using 2-d temperature distributions for the top fuel node. (This option is not yet available.)

1296 ROGSPI

Mass of fission gas generated in the fuel pin per unit volume of the original pin and percent burnup. Used to set the fission gas arrays only if DEFORM is not used (**ISSFUE** = 0).

1297 PRSFTN

Partial pressure of dissolved gas, due to surface tension. This pressure is calculated as 2 x SIGMA/R. The recommended value for oxide fuel is 320 x 10**5 Pa. (SIGMA = 400 x 10^{-3} N/m, R = 250 x 10**-10 m).

1298 COLFAC

Multiplier on the sodium density in the coolant reactivity calculation. Suggested value: 1.0.

1299 GAMGAS

Cp/Cv for pin plenum gas, P*V**GAMGAS = constant for adiabatic changes. Typical values: 1.3 - 1.6.

METAL FUEL PROPERTIES DATA

1300-1315 PUZRTP (L,IFUEL)

Metal fuel plutonium and zirconium weight fractions by fuel type (IFUEL). Used for **IMETAL** > 1. L=1, Plutonium weight fraction. L=2, Zirconium weight fraction.

1316-1323 RHOZN (IFUEL)

Metal fuel theoretical density at the reference temperature (TR) by fuel type (IFUEL). Used for **IMETAL** > 1. If **RHOZN** = -1.0, then it is internally evaluated based on the input composition (**PUZRTP**). If **RHOZN** = 0.0, then the fuel type IFUEL does not use the IFR Handbook-interpolated U-Pu-Zr alloy fuel properties of input option **IFUELM** = 0.

1324-1331 **XLOGNA (IFUEL)**

Metal fuel porosity fraction logged by bond sodium by fuel type (IFUEL). Used for **IMETAL** > 1.

1332 **XSIGMC**

To account for uncertainties in IFR handbook data for U-Pu-Zr and U-Zr alloy fuel heat capacity, add |**XSIGMC**| standard deviations (subtract if **XSIGMC** is negative) to the best estimate.

1333 **XSIGMK**

To account for uncertainties in IFR handbook data for U-Pu-Zr and U-Zr alloy fuel thermal conductivity, add |**XSIGMK**| standard deviations (subtract if **XSIGMK** is negative) to the best estimate. The standard deviation depends on fuel burnup (See **BURNFU**).

1334 **XSIGMD**

To account for uncertainties in IFR handbook data for U-Pu-Zr and U-Zr fuel theoretical density, add |**XSIGMD**| standard deviations (subtract if **XSIGMD** is negative) to the best estimate.

1335-1394 **APROPI (I)**

Coefficients for user-defined coolant properties.

APROPI(1:59) corresponds to the A_i defined in the coolant property equations of Section 12.13. **APROPI**(60) normally corresponds to the coolant critical temperature, T_c. See Eq. (12.13-7).

1395-1725 **DUMPMC**

Not currently used.

2.7.8 Block 14 — PRIMIN — Primary Loop Input Data

1 **PX** **Pa**

Coolant exit pressure at **ZPLENU**.

2 **PDEC** **1/s**

Pump head decay constant. (See **PDEC1** and PDEC2).

3 **PDEC1** **1/s^2**

Pump head decays as exp (-**PDEC***t - **PDEC1***t^2 - **PDEC2***t^3).

4 **PDEC2** **1/s^3**

Pump head decays as exp (-**PDEC***t - **PDEC1***t^2 - **PDEC2***t^3).

5-24 **PRETAB (L)**

Normalized inlet coolant driving pressure or coolant flow rate table as a function of **PRETME**. Used only if **IPRION** ≠ 4. The inlet coolant driving pressure is the pressure difference between the subassembly inlet and outlet minus the gravity head across the subassembly. (See **IFLOW**).

25-44 **PRETME (L)** **s**

Time for **PRETAB** table.

45-64 **T0TAB (L)** **K**

Inlet temperature. (See **NT0TAB**).

65-84 **T0TME (L)** **s**

Time for **T0TAB** table.

85 **TNTRY** **K**

Re-entry temperature. Used only if PRIMAR-4 is not used.

86 **DPGRV0** **Pa**

The primary loop gravity head. If 0.0, **DPGRV0** is calculated at steady-state. Used only if **IPRION** ≠ 4.

87 **ZPLENL** **m**

Reference elevation for the coolant inlet plenum.

88 **ZPLENU** **m**

Reference elevation for the coolant outlet plenum.

89 **ZIHX** **m**

Location of the thermal center of the intermediate heat exchanger. Used if **IPRION** ≠ 4 and **DPGRV0** = 0.

90 **DZBCGL** **m**

Minimum distance above subassembly exit for bubble breakaway. Suggested value: 0.1.

91 **DZBCGU** **m**

Maximum distance above subassembly exit for bubble breakaway. Suggested value: 0.25. Note: When the top of the bubble is above **DZBCGU** or the bottom is above **DZBCGL**, the portion above **DZBCGL** is broken away.

92 **TIMDBG** **s**

Time when PRIMAR1 debug prints start.

93 **XMXMSI** **kg**

Coolant mass per pin in the mixing volume at the subassembly inlet.

94 **XMXMSO** **kg**

Coolant mass per pin in the mixing volume at the subassembly outlet. If this quantity varies from channel to channel, then input the value for channel which boils first.

95 **TIMMIX** **s**

Time constant for heat flow between inlet mixing volume and bulk inlet plenum temperature, also between outlet mixing volume and bulk outlet plenum temperature.

96 **THT2VL**

Minimum degree of implicitness for flow calculations in PRIMAR-4 and TSCNV1.

97-150 **DUMPRI**

Not currently used.

DEFAULT VALUES

XMXMSI = 0.002
XMXMSO = 0.002
TIMMIX = 1.00

Note: Variables marked P4 must be input when using PRIMAR-4. Other variables in this block are used for PRIMAR-1.

2.7.9 Block 15 — FINBOP — Balance-of-Plant Floating-Point Input

Note: The first-encountered FINBOP is saved and may not be redefined by subsequent data. Within the block, the data must be presented as defined in ANL/RAS 89/6. This block must follow Block 6, *IINBOP*.

See ANL/RAS 89/6, Appendix B.

The input format for each record in Block 15 is:

```
(2I3,I6,5E12.5)
```

The first integer on the line identifies the type of component for which data are being entered. The numbers run as follows:

Type	Description
1	compressible volume
2	segment
3	element
4	pump
5	simple heater model
6	standard valve
7	check valve
8	flow boundary condition
9	volume boundary condition
10	steam generator
11	open heater
12	condenser
13	reheater
14	flashed heater
15	drain cooler
16	desuperheating heater
17	desuperheater/drain cooler
18	steam drum
19	turbine stage
20	nozzle
21	superheater
22	relief valve.

The second integer is the continuation card number, NCONTN. The third integer is the user's number for the component. The floating point data entries are explained below. The units for each entry are given in parentheses at the end of the entry description.

Volumes

(2I3,I6,5E12.5): **1**, **1**, user's no., VOLCVW, ZCVW, X, Y, Z

VOLCVW: volume of the compressible volume (m^3),

ZCVW: elevation of the midpoint of the volume (m)

X, Y, Z: chosen by the user to be one of the following combinations (the user must set flag NENTRF on fixed point volume card (1,1) to signal to the code which combination is used):

NENTRF = 1:
X = PRESW4,
Y = TCVW,
Z = null,

PRESW4 is the steady state volume pressure (Pa),
TCVW is the steady state volume temperature (K).

NENTRF = 2:
X = PRESW4,
Y = HCVW,
Z = null,

HCVW is the steady state volume specific enthalpy (J/kg).

NENTRF = 3:
X = PRESW4,
Y = XCVW,
Z = null,

XCVW is the quality in the volume.

NENTRF = 4:
X = TCVW,
Y = XCVW,
Z = null.

NENTRF = 5:
X = PRESW4,
Y = TPFACE,
Z = TAMBNT,

TPFACE is the two-phase level in the volume (m),
TAMBNT is the ambient temperature (K).

NENTRF = 6:
X = TCVW,
Y = TPFACE,
Z = TAMBNT.

Segments

(2I3,I6,5E12.5): **2**, **1**, user's no., FLOWSS, ZINW, CVMLTW(1) , CVMLTW(2)

FLOWSS: steady-state flow in the segment (kg/s)

ZINW: elevation of the segment inlet (m)

CVMLTW(1): multiplicity factor at the segment inlet

CVMLTW(2): multiplicity factor at the segment outlet

(See ANL/RAS 89/6, Section 8.4, for an explanation of multiplicity factors. If the nodalization does not use multiplicity factors to take advantage of symmetries in the plant, the multiplicity factors at the ends of all segments must be set to 1.0).

(2I3,I6,5E12.5): **2**, **2**, user's no., SEGTMP(2)

SEGTMP tube-side temperature at the outlet of the tube bundle if the segment passes through a heater. See ANL/RAS 89/10 for more details on this parameter (K).

Elements

(2I3,I6,5E12.5): **3**, **1**, user's no., ZOUTLW, XLENW, AREAW, DHW, G2PW

ZOUTLW: elevation of the element outlet (m),

XLENW: length of the element (m),

AREAW: element cross-sectional area (m^2),

DHW: element hydraulic diameter (m), and

G2PW: orifice coefficient (G2PW is normally set to zero in the input and is then computed by the code, as the orifice coefficient is usually difficult for the user to determine. The user must choose between entering the orifice coefficient or the element endpoint pressure PELEW (See element card (3,2)) for any element for which the outlet does not interface with a compressible volume. Usually, the pressure is much more readily available to use as input data; however, there may be cases in which a user wishes to enter orifice coefficients instead, and so this option has been made available at this point.). G2PW is dimensionless.

(2I3,I6,5E12.5): **3**, **2**, user's no., ROUGHW, BENDW, PELEW

ROUGHW: surface roughness of the element wall (this parameter is used in the expression for the Moody friction factor; see L. F. Moody, Mech. Eng., 69, p. 1005 (1947), for a detailed explanation of surface roughness) (m).

BENDW: number of bends in the element.

PELEW: element endpoint pressure. This quantity is entered only for elements for which the outlet does not interface with a compressible volume (Pa).

The remaining element cards are used exclusively by the detailed heater models; see ANL/RAS 89/10 for details about any of these parameters.

(2I3,I6,5E12.5): **3**, **3**, user's no., TBTHIK, TBRHO, TBCP, TBKPM0, ZLOWST

TBTHIK: tube thickness (m),

TBRHO: density of the tube material (kg/m^3),

TBCP: specific heat of the tube material (J/kg-K),

TBKPM0: thermal conductivity of the tube material (W/m-K),

ZLOWST: lowest elevation of the tube bundle within the heater volume (m).

(2I3,I6,5E12.5): **3**, **4**, user's no., TBNODE, TBNMBR

TBNODE: number of nodes for the heat transfer calculation in the element,

TBNMBR: total number of tubes in the bundle.

(2I3,I6,5E12.5): **3**, **5**, user's no. TBNDLW TBLNLW

TBNDLW: number of nodes for the section of the element within the drain,

TBLNLW: length of the section of the element within the drain (m).

(2I3,I6,5E12.5): **3**, **6**, user's no., TBNDUP, TBLNUP

TBNDUP: number of nodes for the section of the element within the desuperheating section,

TBLNUP: length of the section of the element within the desuperheating section (m).

Pumps

Rated values of the pump parameters:

(2I3,I6,5E12.5): **4**, **1**, user's no., HEADWR, PMPFWR, PMPSWR, PMWEFR, PMWINR

HEADWR: rated pump head (Pa),

PMPFWR: rated pump flow (kg/s),

PMPSWR: rated pump speed (rad/s),

PMWEFR: rated pump efficiency, and

PMWINR: rated pump inertia (kg-m^2).

These definitions apply to the two centrifugal pump models (pump types 1 and 2). If the pump is modeled through a table of user-specified pump head vs. time (pump type 0), only HEADWR need be entered; the other four variables are not applicable.

(2I3,I6,5E12.5): **4**, **2**, user's no., TRKLSW

TRKLSW: windage (the torque coefficient for shaft friction).

This variable applies only to the two centrifugal pump models.

(2I3,I6,5E12.5): **4**, **3**, user's no., APMWHD(1-5)

APMWHD: contains the first five pump head coefficients in the case of the two centrifugal pump models and contains the first half of a table of normalized pump head in the case of user-specified pump head vs. time (pump type 0) or normalized pump head vs. normalized flow (to be used when the locked rotor option -2 has been invoked).

(2I3,I6,5E12.5): **4**, **4**, user's no., APMWHD(6-10)

APMWHD: contains the last five pump head coefficients in the case of the two centrifugal pump models and contains the continuation of a table of normalized pump head in the case of pump type 0 or locked rotor option -2.

(2I3,I6,5E12.5): **4**, **5**, user's no., APMWHD(11-15)

APMWHD: contains the first five pump torque coefficients in the case of the two centrifugal pump models and contains the continuation of the table of normalized pump head in the case of pump type 0 or locked rotor option -2.

(2I3,I6,5E12.5): **4**, **6**, user's no., APMWHD(16-20)

APMWHD: contains the last five pump torque coefficients in the case of the two centrifugal pump models and contains the end of the table of normalized pump head in the case of pump type 0 or locked rotor option -2.

(2I3,I6,5E12.5): **4**, **7**, user's no., WMOTTK(1-5)

WMOTTK: contains the first five entries in the motor torque table in the case of the two centrifugal pump models, the first five time entries for the relative pump head table in the case of pump type 0 (s), and the first five normalized flow entries in the case of locked rotor option -2.

(2I3,I6,5E12.5): **4**, **8**, user's no., WMOTTK(6-10)

WMOTTK: contains the last five entries in the motor torque table in the case of the two centrifugal pump models, the continuation of the time table in the case of pump type 0 (s), and the continuation of the normalized flow table in the case of locked rotor option -2.

(2I3,I6,5E12.5): **4**, **9**, user's no., WMOTTK(11-15)

WMOTTK: contains the first five entries in the table of times for the motor torque table in the case of the centrifugal pump models; the continuation of the time table in the case of pump type 0 (s); and the continuation of the normalized flow table in the case of locked rotor option -2.

(2I3,I6,5E12.5): **4**, **10**, user's no., WMOTTK(16-20)

WMOTTK: contains the last five entries in the table of times for the motor torque table in the case of the centrifugal pump models; the end of the time table in the case of pump type 0 (s); and the end of the normalized flow table in the case of locked rotor option -2.

Simple Heater Model

(2I3,I6,5E12.5): **5**, **1**, user's no., APRMHT, AXPMHT, DHPMHT, TSECHT

APRMHT: heat transfer area between the primary and secondary sides (m^2),

AXPMHT: primary side flow area (m^2),

DHPMHT: primary side hydraulic diameter (m), and

TSECHT: secondary side temperature (K).

The user's number for the heater is not the compressible volume number but rather the number entered in array NQFLG.

Standard Valve

(2I3,I6,5E12.5): **6**, **1**, user's no., VLVMSW, VDAMPW, VSPRGW, VSTEMW, VCONSW

VLVMSW: valve mass (kg),

VDAMPW: valve damping coefficient (kg-m/s),

VSPRGW: valve spring constant (kg/s^2),

VSTEMW: initial valve stem position (m)

VCONSW: constant of proportionality between the stem position and the valve characteristic.

(2I3,I6,5E12.5): **6**, **2**, user's no., VPHIW(1-5)

(2I3,I6,5E12.5): **6**, **3**, user's no., VPHIW(6-10)

VPHIW: valve coefficient table.

(2I3,I6,5E12.5): **6**, **4**, user's no., VPOSW(1-5)

(2I3,I6,5E12.5): **6**, **5**, user's no., VPOSW(6-10)

VPOSW: is the valve stem position table (m).

(2I3,I6,5E12.5): **6**, **6**, user's no., VTABDW(1-5)

(2I3,I6,5E12.5): **6**, **7**, user's no., VTABDW(6-10)

VTABDW: table of valve driver function ($kg-m/s^2$).

(2I3,I6,5E12.5): **6**, **8**, user's no., VTIMW(1-5)

(2I3,I6,5E12.5): **6**, **9**, user's no., VTIMW(6-10)

VTIMW: table of time corresponding to VTABDW (s).

Check Valve

(2I3,I6,5E12.5): **7**, **1**, user's no., CHPHIW, CHEPS1, CHEPS2

CHPHIW: valve characteristic when the valve is fully open,

CHEPS1: criterion for starting to close an open valve (Pa or kg/s, depending on whether the user has chosen to initiate valve closure on a pressure drop criterion or a flow criterion),

CHEPS2: criterion for starting to open a closed valve (Pa or kg/s, depending on whether the user has chosen to initiate valve opening on a pressure drop criterion or a flow criterion).

(2I3,I6,5E12.5): **7**, **2**, user's no., CVPHIC(1-5)

CVPHIC: normalized valve coefficient table for a valve which is closing.

(2I3,I6,5E12.5): **7**, **3**, user's no., CVPHIC(6-10)

(2I3,I6,5E12.5): **7**, **4**, user's no., CVTIMC(1-5)

CVTIMC: array of time entries corresponding to CVPHIC (s).

(2I3,I6,5E12.5): **7**, **5**, user's no., CVTIMC(6-10)

(2I3,I6,5E12.5): **7**, **6**, user's no., CVPHIO(1-5)

CVPHIO: normalized valve coefficient table for a valve which is opening.

(2I3,I6,5E12.5): **7**, **7**, user's no., CVPHIO(6-10)

(2I3,I6,5E12.5): 7, **8**, user's no., CVTIMO(1-5)

CVTIMO: array of time entries corresponding to CVPHIO (s).

(2I3,I6,5E12.5): **7**, **9**, user's no., CVTIMO(6-10)

Flow Boundary Condition

(2I3,I6,5E12.5): **8**, **1**, table no., entry no., TABSEG(1), TABSEG(2), X, Y

where the entry number indicates which entry is being made in this flow boundary condition table,

TABSEG(1): time (s),

TABSEG(2): absolute (not normalized) flow (kg/s),

X, Y: any of the following combinations for an inflow boundary condition (the flag NFLSEG on fixed point volume card (1,1) designates which combination is chosen):

NFSLEG = 0:
X = specific enthalpy (J/kg),
Y = 0,

NFLSEG = 1:
X = temperature (K),
Y = pressure (Pa) for a subcooled liquid boundary condition,

NFLSEG = 2:
X = temperature (K),
Y = pressure (Pa) for a superheated vapor boundary condition,

NFLSEG = 3:
X = quality,
Y = pressure (Pa),

NFLSEG = 4:
X = quality,
Y = temperature (K).

X and Y are not required for an outflow boundary condition.

Volume Boundary Condition Card

(2I3,I6,5E12.5): **9**, **1**, table no., entry no., TABVOL(1), X, Y

where the entry number indicates which entry is being made in this flow boundary condition table.

TABVOL(1): time (s),

X, Y: any of the following combinations (the flag NTABVL on volume boundary condition card (6,1) designates which combination is chosen):

NTABVL = 1:
X = pressure (Pa),
Y = specific enthalpy (J/kg) for a subcooled liquid volume,

NTABVL = 2:
X = pressure (Pa),
Y = temperature (K) for a subcooled liquid volume,

NTABVL = 3:
X = pressure (Pa),
Y = specific enthalpy (J/kg) for a superheated vapor volume,

NTABVL = 4:
X = pressure (Pa),
Y = temperature (K) for a superheated vapor volume,

NTABVL = 5:
X = pressure (Pa),
Y = quality,

NTABVL = 6:
X = temperature (K),
Y = quality.

Steam Generator

(2I3,I6,5E12.5): **10**, **1**, steam gen. number, DEWI, ZONLE(1), ZONLE(3), XKTUBE, COILD

DEWI: booster tube outer diameter (m),

ZONLE(1): initial subcooled zone length (m),

ZONLE(3): initial superheated zone length (m),

XKTUBE: tube thermal conductivity (W/m-K), and

COILD: average diameter of the coil in the helical coil geometry option for the evaporator/steam generator model.

(2I3,I6,5E12.5): **10**, **2**, steam gen. number, HTFI(1), HTFI(2), HTFI(3), HTFI(4), VRISE

HTFI(I): calibration factors for heat transfer coefficients for each regime,

VRISE: vertical rise per length of helical tube in the helical coil geometry option for the evaporator/steam generator model.

(2I3,I6,5E12.5): **10**, **3**, steam gen. number, ROCPTB, FOULRI(1), FOULRI(2), FOULRI(3) FOULRI(4)

ROCPTB: reserved,

FOULRI(I): fouling heat resistances on the water side for each heat transfer regime.

(2I3,I6,5E12.5): **10**, **4**, steam gen. number, PICHT, PICHL

PICHT: transverse pitch of the helical tubes in the helical coil geometry option for the evaporator/steam generator model,

PICHL: longitudinal pitch of the helical tubes in the helical coil geometry option for the evaporator/steam generator model.

(See Block 18, #3934-4173 for the remainder of the steam generator input).

Deaerator

(2I3,I6,5E12.5): **11**, **1**, user's volume number, QRATIO, HTRELV, HTRRAD

QRATIO: percentage of incoming energy lost due to imperfect insulation,

HTRELV: elevation of the lowest point of the heater (m),

HTRRAD: heater radius (m).

Condenser

(2I3,I6,5E12.5): **12**, **1**, user's volume number, QRATIO, HTRELV, HTRRAD, CSAREA, DHSHW

QRATIO: percentage of incoming energy lost due to imperfect insulation,

HTRELV: elevation of the lowest point of the heater (m),

HTRRAD: heater radius (m),

CSAREA: effective cross-sectional area of the heater (m^2),

DHSHW: hydraulic diameter of the shell side of the heater (m).

(2I3,I6,5E12.5): **12**, **2**, user's volume number, SHHTCC

SHHTCC: shell side condensation coefficient (W/m-K).

Reheater

(2I3,I6,5E12.5): **13**, **1**, user's volume number, QRATIO, HTRELV, HTRRAD, CSAREA, DHSHW

QRATIO: percentage of incoming energy lost due to imperfect insulation,

HTRELV: elevation of the lowest point of the heater (m),

HTRRAD: heater radius (m),

CSAREA: effective cross-sectional area of the heater (m^2)

DHSHW: hydraulic diameter of the shell side of the heater (m).

(2I3,I6,5E12.5): **13**, **2**, user's volume number SHHTCC

SHHTCC: shell side condensation coefficient (W/m-K).

Flashed Heater

(2I3,I6,5E12.5): **14**, **1**, user's volume number, QRATIO, HTRELV, HTRRAD, CSAREA, DHSHW,

QRATIO: percentage of incoming energy lost due to imperfect insulation,

HTRELV: elevation of the lowest point of the heater (m),

HTRRAD: heater radius (m),

CSAREA: effective cross-sectional area of the heater (m^2),

DHSHW: hydraulic diameter of the shell side of the heater (m).

(2I3,I6,5E12.5): **14**, **2**, user's volume number, SHHTCC

SHHTCC: shell side condensation coefficient (W/m-K),

Drain Cooler

(2I3,I6,5E12.5): **15**, **1**, user's volume number, QRATIO, HTRELV, HTRRAD, CSAREA, DHSHW

QRATIO: percentage of incoming energy lost due to imperfect insulation,

HTRELV: elevation of the lowest point of the heater (m),

HTRRAD: heater radius (m),

CSAREA: effective cross-sectional area of the heater (m^2),

DHSHW: hydraulic diameter of the shell side of the heater (m).

(2I3,I6,5E12.5): **15**, **2**, user's volume number, SHHTCC, HIGHLW, XLENLW

SHHTCC: shell side condensation coefficient (W/m-K),

HIGHLW: height of the drain (m),

XLENLW: length of the drain (m).

(2I3,I6,5E12.5): **15**, **3**, user's volume number, ORIFLW, TEMPLW, VOLLW, CSARLW, DHLWW, , ,

ORIFLW: elevation of the drain orifice (m),

TEMPLW: drain temperature (K),

VOLLW: drain volume (m^3),

CSARLW: effective cross-sectional area of the drain (m^2),

DHLWW: hydraulic diameter of the drain (m).

Desuperheating Heater

(2I3,I6,5E12.5): **16**, **1**, user's volume number, QRATIO, HTRELV, HTRRAD, CSAREA, DHSHW

QRATIO: percentage of incoming energy lost due to imperfect insulation,

HTRELV: elevation of the lowest point of the heater (m),

HTRRAD: heater radius (m),

CSAREA: effective cross-sectional area of the heater (m^2),

DHSHW: hydraulic diameter of the shell side of the heater (m).

(2I3,I6,5E12.5): **16**, **2**, user's volume number, SHHTCC, HIGHUP, XLENUP

SHHTCC: shell side condensation coefficient (W/m-K),

HIGHUP: height of the desuperheating region (m),

XLENUP: length of the desuperheating region (m).

(2I3,I6,5E12.5): **16**, **3**, user's volume number, ORIFUP, TEMPUP, VOLUP, CSARUP, DHUPW

ORIFUP: elevation of the desuperheating region orifice (m),

TEMPUP: desuperheating region temperature (K),

VOLUP: desuperheating region volume (m^3),

CSARUP: effective cross-sectional area of the desuperheating region (m^2),

DHUPW: hydraulic diameter of the desuperheating region (m).

Desuperheater/Drain Cooler

(2I3,I6,5E12.5): **17**, **1**, user's volume number, QRATIO, HTRELV, HTRRAD, CSAREA, DHSHW,

QRATIO: percentage of incoming energy lost due to imperfect insulation,

HTRELV: elevation of the lowest point of the heater (m),

HTRRAD: heater radius (m).

CSAREA: effective cross-sectional area of the heater (m^2),

DHSHW: hydraulic diameter of the shell side of the heater (m),

(2I3,I6,5E12.5): **17**, **2**, user's volume number, SHHTCC, HIGHLW, XLENLW, HIGHUP, XLENUP

SHHTCC: shell side condensation coefficient (W/m-K),

HIGHLW: height of the drain (m),

XLENLW: length of the drain (m),

HIGHUP: height of the desuperheating region (m),

XLENUP: length of the desuperheating region (m).

(2I3,I6,5E12.5): **17**, **3**, user's volume number, ORIFLW, TEMPLW, VOLLW, CSARLW, DHLWW

ORIFLW: elevation of the drain orifice (m),

TEMPLW: drain temperature (K),

VOLLW: drain volume (m^3),

CSARLW: effective cross-sectional area of the drain (m^2)

DHLWW: hydraulic diameter of the drain (m).

(2I3,I6,5E12.5): **17**, **4**, user's volume number, ORIFUP, TEMPUP, VOLUP, CSARUP, DHUPW, , ,

ORIFUP: elevation of the drain orifice (m),

TEMPUP: drain temperature (K),

VOLUP: drain volume (m^3),

CSARUP: effective cross-sectional area of the drain, (m^2),

DHUPW: hydraulic diameter of the drain (m).

Steam Drum

(2I3,I6,5E12.5): **18**, **1**, user's volume number, QRATIO, HTRELV, HTRRAD

QRATIO: percentage of incoming energy lost due to imperfect insulation,

HTRELV: elevation of the lowest point of the heater (m),

HTRRAD: heater radius (m).

Turbine Stage

(2I3,I6,5E12.5): **19**, **1**, user's volume number, RROTER, (flag for last stage), TRGRMI, OMEGAR

RROTOR: rotor radius (m),

(flag for last stage) is a flag which is zero unless this volume is the last stage of the turbine; for the last stage of the turbine, the user must enter any positive number for this flag.

TRGRMI: turbine/generator rotor moment of inertia (kg-m^2),

OMEGAR: rotor angular velocity (rad/s).

Nozzle

(2I3,I6,5E12.5): **20**, **1**, user's segment number, CNNZCF, CRRXCF, CBBKCF, ALFANZ, GAMABL

CNNZCF: nozzle velocity coefficient,

CRRXCF: reactor coefficient,

CBBKCF: bucket coefficient,

ALFANZ: nozzle angle, the angle at which steam enters the turbine blade system (rad),

GAMABL: blade exit angle, the angle at which steam leaves the turbine blade system (rad).

(2I3,I6,5E12.5): **20**, **2**, user's segment number, CONSK1 CONSK2 CONSK3 XRXFR

CONSK1: rotation loss coefficient,

CONSK2: moisture loss coefficient,

CONSK3: exhaust loss coefficient,

XRXFR: reactor fraction, the fraction of stage energy released in the bucket system.

Superheater

(2I3,I6,5E12.5): **21**, **1**, user's number, DEWOS, DEWIS, DOUTS, DHNAS, ARNAS

DEWOS: steam tube inner diameter (m),

DEWIS: booster tube outer diameter (m),

DOUTS: steam tube outer diameter (m),

DHNAS: sodium hydraulic diameter per tube (m),

ARNAS: sodium flow area per tube (m^2).

(2I3,I6,5E12.5): **21**, **2**, user's number, TUBNOS, TBPODS, XKTUBS, COILDS, VRISES

TUBNOS: number of superheater tubes,

TBPODS: superheater bundle pitch-to-diameter ratio,

XKTUBS is reserved,

COILDS: average diameter of the coil for the helical coil geometry option in the superheater model,

VRISES: vertical rise per length of helical tube in the helical coil geometry option in the superheater model.

(2I3,I6,5E12.5): **21**, **3**, user's number, FOULSI, PICHTS, PICHLS

FOULSI: fouling heat resistance on the water side,

PICHTS: transverse pitch of the helical tube in the helical coil geometry option for the superheater model,

PICHLS: longitudinal pitch of the helical tube in the helical coil geometry option for the superheater model.

Relief Valve

(2I3,I6,5E12.5): **22**, **1**, user's relief valve number, RVA, DPBLD, DPSET, DPACC, RVFRAC

RVA: fractional valve area to which the valve opens when the set pressure drop is reached,

DPBLD: blowdown pressure drop, at which a partially open relief valve shuts (Pa),

DPSET: set pressure drop, at which a closed relief valve starts to open (Pa),

DPACC: accumulated pressure drop, at which the valve is completely open (Pa),

RVFRAC: fractional valve opening area (0 for a fully closed valve, 1 for a fully open valve).

(2I3,I6,5E12.5): **22**, **2**, user's relief valve number, PCVI, PCVO, HCVI, FLOWLS, TIMERV

PCVI: initial pressure upstream of the valve (Pa),

PCVO: initial pressure downstream of the valve (Pa),

HCVI: initial enthalpy upstream of the valve (J/kg),

FLOWLS: valve capacity at the accumulated pressure drop (kg/s),

TIMERV: delay time for opening or closing the valve; the delay time allows the code to avoid numerical problems with the discontinuities in the fractional valve opening area in the valve hysteresis curve of opening area versus valve pressure drop (s).

2.7.10 Block 18 — PMR4IN — PRIMAR-4 Floating-Point Input

1 **TMDBP4** **s**

Time when **IDBPR4** is set to **IDBP4N**. (See **IDBPR4** and **IDBP4N**).

2-41 **FLOSSL (ISGL)** **kg/s**

Initial flow rate in liquid flow segments. The value input for the single segment representing all channels (except the bypass channel) is not used.

42-81 **ZINL (ISGL)** **m**

Height of inlet to liquid segment.

82-161 **CVLMLT (M,ISGL)**

Flow rate multiplicity factors at liquid segment ends. M=1 at inlet, M=2 at outlet.

162-301 **ZOUTEL (IELL)** **m**

Height at outlet of the liquid element. Note: **ZOUTEL**(IELL) is also the height at the inlet of element IELL+1 if IELL+1 is in the same segment.

302-441 **XLENEL (IELL)** **m**

Length of element. Must be greater than zero. (See comment for all channels in **FLOSSL**).

442-581 **AREAEL (IELL)** **m^2**

Cross-sectional flow area of liquid elements. Must be greater than zero. (See comment for all channels in **FLOSSL**).

582-721 **DHELEM (IELL)** **m**

Hydraulic diameter of liquid element. Must be greater than zero.

722-861 **ROUGHL (IELL)** **m**

Pipe surface roughness, for friction factor. Must be greater than or equal to zero.

862-1001 **BENDNM (IELL)**

Number of bends in each liquid element. Must be greater than or equal to zero.

1002-1141 **G2PRDR (IELL)**

Initial orifice coefficient. Must be greater than or equal to zero. G^2 pressure-drop coefficient. ΔP=**G2PRDR***G*ABS(G)/(2.*RHO).

1142 **BNDLOD**

Effective L/D per bend.

1143-1282 **WALLMC (IELL)** **J/m-K**

Pipe wall mass * heat capacity /length. Must be greater than zero.

1283-1422 **WALLH (IELL)** **W/m^2-K**

Pipe wall heat-transfer coefficient. Must be greater than zero.

1423-1460 **VOLLGC (ICV)** **m^3**

Total volume of the compressible volume, liquid + gas.

1461-1498 **PRESG0 (ICV)** **Pa**

Initial gas pressure in the compressible volume having gas.

1499-1536 **ALPHAP (ICV)** **1/Pa**

Volume pressure expansion coefficient, (1/V)dV/dP, for the compressible volume.

1537-1574 **ALPHAT (ICV)** **1/K**

Volume thermal expansion coefficient, (1/V)dV/dT, for the compressible volume.

1575-1612 **ZCVL (ICV)** **m**

Reference height for liquid pressure in the compressible volume.

1613-1650 **AREAIN (ICV)** **m^2**

Area of liquid-gas interface in the compressible volume. Input 1.0 if there is no interface.

1651-1688 **TREFCV (ICV)** **K**

Initial steady-state gas temperature if gas is present. Steady-state gas temperature = liquid temperature if input as 0. Not used for a liquid-only compressible volume.

1689 **GAMGSC**

Cp/Cv for cover gas (P*V)**GAMGSC** = constant. Used for all compressible volumes containing cover gas.

1690 **RGASC** **Pa-m^3/kg-K**

Gas constant for cover gas, P * V = M * **RGASC** * T. Used for all compressible volumes containing cover gas.

R = 8314/molecular weight.

R = 2078. for He.

R = 208.1 for Argon.

1691 **U0CVGS** **Pa-s**

Viscosity of cover gas at reference temperature **TRFU**.

Suggested value:

1.94×10^{-5} at 293 K for He

2.22×10^{-5} at 293 K for Argon

1692 **TRFU** **K**

Reference temperature for gas viscosity.

1693-1720 **XLENGG (ISGG)** **m**

Length of gas segment.

1721-1748 **AREASG (ISGG)** **m^2**

Area.

1749-1804 **CVGMLT (M,ISGG)**

Multiplicity factors at gas segment ends. M=1 at inlet, 2 at outlet. Inlet and outlet the same as for **JCVG**(M,ISGG).

1805-1832 **DHSEGG (ISGG)** **m**

Hydraulic diameter.

1833-1860 **ROUGHG (ISGG)** **m**

Surface roughness.

1861-1898 **TAUGAS (ICV)** **s**

Cover-gas temperature time constant.

PUMP OPTIONS

IEMPMP	ILRPMP	Pump Option
Any	-2	*User-specified normalized pump head vs. normalized flow*.
-3	Any > 0	*Equivalent Circuit EM pump*.
-2	≥ 0	*ALMR EM Pump*.
-1	Any ≠ -2	*Electromagnetic (EM) pump*.
0	Any ≠ -2	*Table look-up, user specified normalized pump head vs. time*.
1	-1	*Centrifugal pump option 1, user-specified normalized pump speed vs. time*.
1	0	*Centrifugal pump option 1, user-specified normalized motor torque vs. time*.
1	1	*Centrifugal pump option 1, locked rotor (pump speed = 0)*.
2	-1	*Centrifugal pump option 2, ANL pump model, user-specified normalized pump speed vs. time*.
2	0	*Centrifugal pump option 2, ANL pump model, user-specified normalized motor torque vs. time*.
2	1	*Centrifugal pump option 2, ANL pump model, locked rotor (pump speed = 0)*.
3	Any ≠ -2	*EBR-II pump model*.
		Note: See **IEMPMP** and **ILRPMP**.

USER-SPECIFIED PUMP HEAD VS TIME

$$(\text{IEMPMP}\,(\text{IPMP}) = 0)$$

1899-1910 **NONE**

Not used.

1911-1922 **HEADR (IPMP)** **Pa**

Pump head at t = 0, computed by the code unless steady-state flow = 0.

1923-1982 **NONE**

Not used.

1983-2222 **APMPHD (J,IPMP)**

Table of relative pump head. Dimension (20, 12).

2223-2462 **AMOTTK (J,IPMP)** **s**

Times for pump head table. Dimension (20, 12).

CENTRIFUGAL PUMP

$$(\text{IEMPMP}\,(\text{IPMP}) = 1)$$

1899-1910 **PMPINR (IPMP)** **kg-m^2**

Moment of inertia, pump and motor.

1911-1922 **HEADR (IPMP)** **Pa**

Rated pump head.

1923-1934 **PMPSPR (IPMP)** **Rad/s**

Rated pump speed.

1935-1946 **PMPFLR (IPMP)** **kg/s**

Rated pump flow.

1947-1958 **PMPEFR (IPMP)**

Rated pump efficiency.

1959-1970 **TRKLSC (IPMP)**

Torque coefficient for shaft friction, windage.

1971-1982 **EPSCAV (IPMP)**

Cavitation coefficient = ANPSH/HEAD for cavitation, rated value.

1983-2222 **APMPHD (IPMP)**

Pump head coefficients, J = 1 to 10, and pump torque coefficients, J = 11 to 20.

2223-2462 **AMOTTK (IPMP)** **s**

Motor torque table, J = 1 to 10, and times for motor torque table, J = 11 to 20.

HOMOLOGOUS PUMP MODEL

$$(\text{IEMPMP}\,(\text{IPMP}) = 2)$$

1899-1910 **PMPINR (IPMP)** **kg-m^2**

Moment of inertia, pump and motor.

1911-1922 **HEADR (IPMP)** **Pa**

Rated pump head.

1923-1934 **PMPSPR (IPMP)** **Rad/s**

Rated pump speed.

1935-1946 **PMPFLR (IPMP)** **kg/s**

Rated pump flow.

1947-1958 **PMPEFR (IPMP)**

Rated pump efficiency.

1959-1970 **TRKLSC (IPMP)**

Torque coefficient for shaft friction, windage.

1971-1982 **EPSCAV (IPMP)**

Cavitation coefficient = ANPSH/HEAD for cavitation, rated value.

1983-2222 **APMPHD (K, IPMP)**

K	Comments
1-3	Torque loss coefficients A(K), low range for APMPHD(10,IPMP) ≥ SB Friction torque = A(1)+A(2)*SB+A(3)*SB^2 SB = normalized pump speed.
4-6	Torque loss coefficients, middle range for APMPHD(11,IPMP) ≥ SB > APMPHD(10,IPMP)
7-9	Torque loss coefficients, high range for SB > APMPHD(11,IPMP)
10-11	Limits of torque loss ranges.
12	Locked rotor loss coefficient.
13	Transition WB, normalized flow for transition from turbulent to laminar pump head.
14	Locked rotor loss coefficient for reverse flow.
15-17	Not used.
18	Pump curve number (1-3).
19	WB for locked rotor.
20	SB for locked rotor. ILRPMP(IPMP) is set to 1 if WB < APMPHD(19,IPMP) and SB < APM-PHD(20,IPMP).

2223-2462 **AMOTTK (K,IPMP)**

Normalized motor torque table, K = 1 to 10, and times for motor torque table, K = 11 to 20. Note: See also **IEMPMP**, **ILRPMP**, **PMPHD**, and **PMPTQ**.

EBR-II PUMP MODEL

$$(\text{IEMPMP}\,(\text{IPMP}) = 3)$$

1983-2222 **APMPHD (J,IPMP)**

Pump head coefficients. Same input format as for the homologous pump model.

2223-2462 **AMOTTK**

Table of normalized pump speed, j = 1 to 10, and times for pump speed, j = 11 to 20.

Note: For an initial steady-state run, the normalization of the speeds in the AMOTTK table does not matter; the code renormalizes the table so that the speeds are relative to the rated speed (PMPSPR). If **AMOTTK** is input for a restart, then it must be normalized to the rated speed.

ELECTROMAGNETIC PUMP

$$(\text{IEMPMP}\,(\text{IPMP}) = -1)$$

$$\text{Pump Head} = \text{Stall Head}\,(t) \times \left[1 - \left(\frac{\text{Flow}}{\text{Synchronous Flow}}\right)^{B}\right]$$

1899-1910 **NONE**

Not used.

1911-1922 **HEADR (IPMP)** **Pa**

Stall head at $t = 0$, computed by the code if steady-state flow is not zero.

1923-1934 **NONE**

Not used.

1935-1946 **PMPFLR (IPMP)** **s/m**

1/Synchronous velocity.

1947-1958 **PMPEFR (IPMP)**

Coefficient B. Suggested value: 1.30.

1959-1982 **NONE**

Not used.

1983-2222 **APMPHD (J,IPMP)**

Table of normalized stall head.

2223-2462 **AMOTTK (J,IPMP)** **s**

Times for **APMPHD** table.

ALMR EM PUMP

$$(\text{IEMPMP}\,(\text{IPMP}) = -2)$$

EM Pump with motor-generator to extend the coastdown time, see formulation in Section 5.3.4.3.2.

1899-1910 **PMPINR (IPMP)** **kg-m^2**

Moment of inertia, I, in Eq. (5.3-161).

1911-1922 **HEADR (IPMP)** **Pa**

Rated pump head, H_{r}. See Eq. (5.3-152).

1923-1934 **PMPSPR (IPMP)** **Rad/s**

Rated frequency, or equivalently, pump speed, f_{r}. See Eq. (5.3-154).

1935-1946 **PMPFLR (IPMP)** **kg/s**

Rated mass flow rate, w_{r}. See Eq. (5.3-155).

1947-1958 **PMPEFR (IPMP)**

Rated pump efficiency, $\varepsilon_{\mathrm{fr}}$. See Eq. (5.3-157).

1959-1982 **TRKLSC (IPMP)**

Torque friction loss coefficient L_{m}. See Eq. (5.3-164).

1983-2222 **APMPHD (J,IPMP)**

ALMR EM Pump model description.

J	Description	Units
1-5	Coefficient a_j in head curve given by Eq. (5.3-156). See defaults in Table 5.3.5, which are specific to the ALMR design.	Dimensionless
6	Friction loss coefficient L_f in Eq. (5.3-151).	Dimensionless
7	Time to cut-over to the motor-generator, i.e. trip time. If **ILRPMP**>0, then the time specified here is ignored and replaced with the value returned by the associated Function Block.	s
8	f_0, frequency prior to cutting over to the motor-generator.	rad/s
9	V_r, rated voltage. See Eq. (5.3-153).	V
10-12	Not used	N/A
13	V_{fr}, cut-over voltage fraction. See Eq. (5.3-160).	Dimensionless
14-20	Not used	N/A

2223-2462 **AMOTTK (J,IPMP)**

ALMR EM Pump model description, continued.

J	Description	Units
1-7	Coefficient b_j in Eq. (5.3-158). See defaults in Table 5.3.5, which are specific to the ALMR design.	Dimensionless
8-16	Coefficient c_{j-7} in Eq. (5.3-159). See defaults in Table 5.3.5, which are specific to the ALMR design.	Dimensionless

EQUIVALENT CIRCUIT EM PUMP

The PMR4IN block input locations are not used for entering Equivalent Circuit EM Pump descriptions. Instead, see description of *EMPUMP blocks*.

NORMALIZED PUMP HEAD VS. NORMALIZED FLOW

$$(\text{ILRPMP}\,(\text{IPMP}) = -2)$$

1899-1982 **NONE**

Not used.

1983-2222 **APMPHD (J,IPMP)**

Table of normalized pump head.

2223-2462 **AMOTTK (J,IPMP)**

Normalized flow for the **APMPHD** table.

2463 **GRAVTY** **m/s^2**

Acceleration of gravity. Preferred input location is **GRAVITY**. **GRAVTY** will be used for the core channels, PRIMAR-4, and BOP if **GRAVITY** is not defined. If neither **GRAVITY** or **GRAVTY** are defined, the default value is 9.81 m/s^2.

2464-2501 **BTAPNA (ICV)** **1/Pa**

Coolant isothermal compressibility, (1/RHO) d RHO/dP, for the compressible volume. Suggested value: 2.13E-10 for sodium at 720 K.

2502-2539 **BTATNA (ICV)** **1/K**

Coolant thermal expansion coefficient, (1/RHO) d RHO/dT, for the compressible volume. Suggested value: -2.8E-4 for sodium at 720 K.

2540-2577 **RHONAR (ICV)** **kg/m^3**

Reference density of Na at **TREFCV**. Default: 844. Not currently used.

2578-2615 **HWALL (ICV)** **W/m^2-K**

Wall-coolant heat-transfer coefficient for the compressible volume.

2616-2653 **AWALL (ICV)** **m^2**

Wall surface area for the compressible volume.

2654-2691 **CMWALL (ICV)** **J/K**

Mass x specific heat for the wall of the compressible volume.

2692 **VOLMIX**

Not currently used.

2693 **TAUUP**

Not currently used.

2694 **C1BY**

Coefficient in Lyon-Martinelli convective heat-transfer correlation for bypass channel. First coefficient set.

(See **C1**, **C1BY2**, **C1BY3**, **C1BY4**, **IHTBYB**, and **IHTBYD**).

2695 **C2BY**

See **C1BY**.

2696 **C3BY**

See **C1BY**.

2697-2704 **XKALBY (IBYP)** **W/m-K**

Thermal conductivity of bypass wall A, lower part. 1 ≤ IBYP ≤ **NBYP**.

2705-2712 **XKAUBY (IBYP)** **W/m-K**

Thermal conductivity of bypass wall A, upper part.

2713-2720 **XKBLBY (IBYP)** **W/m-K**

Thermal conductivity of bypass wall B, lower part.

2721-2728 **XKBUBY (IBYP)** **W/m-K**

Thermal conductivity of bypass wall B, upper part.

2729-2736 **XKDBY (IBYP)** **W/m-K**

Thermal conductivity of bypass wall D.

2737-2744 **DABY (IBYP)** **m**

Thickness of bypass wall A.

2745-2752 **DBBY (IBYP)** **m**

Thickness of bypass wall B.

2753-2760 **DDBY (IBYP)** **m**

Thickness of bypass wall D.

2761-2768 **RCALBY (IBYP)** **J/m^3-K**

Density * specific heat of bypass wall A, lower part.

2769-2776 **RCAUBY (IBYP)** **J/m^3-K**

Density * specific heat of bypass wall A, upper part.

2777-2784 **RCBLBY (IBYP)** **J/m^3-K**

Density * specific heat of bypass wall B, lower part.

2785-2792 **RCBUBY (IBYP)** J/m^3-K

Density * specific heat of bypass wall B, upper part.

2793-2800 **RCDBY (IBYP)** J/m^3-K

Density * specific heat of bypass wall D.

2801-2856 **PSHPBY (J,IBYP)**

Power shape by nodes for each bypass channel. Dimension (7,8). Code normalizes distribution. For the number of nodes, see **NTNODE**.

2857-2880 **GAMNBY (IREG,IBYP)**

Fraction of total reactor power by region for each bypass channel. If there are no nodes in region 2, then set **GAMNBY**(2,IBYP) to 0. Dimension (3,8). Note: See **FRPR**. Sum over all of the three regions of all the **NBYP** bypass channels of the array elements of **GAMNBY** = 1 - **FRPR**.

2881-2904 **GAMDBY (IREG,IBYP)**

Fraction of decay power by region for each bypass channel. Set to 0. The code adds **GAMNBY** to **GAMDBY** for the total power fraction. Dimension (3,8).

2905-2912 **PRFRA1 (IBYP)**

The fraction of the reactor power distributed in reflector A in region 1 of the bypass channel.

2913-2920 **PRFRA2 (IBYP)**

The fraction of the reactor power distributed in reflector A in region 2 of the bypass channel.

2921-2928 **PERABY (IBYP)** m

Perimeter of wall B in bypass channel.

2929-2936 **PERDBY (IBYP)** m

Perimeter of wall D in bypass channel.

2937-3104 **DTMPTB (K,ITAB)**

Table of valve pressure drop coefficient (See **ITABVV**).

Table of normalized temperature drop for an IHX (See **IHXCLC**) or steam generator (See **ISGCLC**).

Table of primary side outlet temperature for an IHX (See **IHXCLC**) or steam generator (See **ISGCLC**). **DTMPTB**(1,ITAB)>1.0

Table of normalized heat rejection for an IHX (See **IHXCLC**) or steam generator (See **ISGCLC**). **DTMPTB**(1,ITAB)=1.0. **VSIHX1**, **VSIHX2**, and **DTSIHX** must be nonzero.

Dimension(14,12).

3105-3272 **ZCENTR (K,ITAB)** **m**

Height of thermal center for TABLE(ITAB) for an IHX or steam generator.

3273-3440 **TMPMTB (K,ITAB)** **s**

Times for **DTMPTB** and **ZCENTR**.

3441-3444 **DTEVPF (IIHX)**

Fraction of total steam generator Na temperature drop in evaporator for the intermediate loop connected to IHX(IIHX). Will be reset to 0.7 if equal to 0. or 1. Suggested value: 0.9999 for table lookup option.

3445-3448 **PRSIHX (IIHX)** **Pa**

Steady-state intermediate side IHX inlet pressure.

3449-3452 **PRSDRC (IDRC)** **Pa**

Steady-state intermediate side DRACS inlet pressure.

Not implemented. See instead **ISGLNK** and **ICVNAK** and use **PRSIHX**.

3453 **PINMIN** **Pa**

Minimum subassembly inlet pressure. Default: 10000.

3454 **DMP4I1**

Not currently used.

3455-3515 **DZSHPX (IDZIHX)** **m**

Fractional axial node heights in IHX.

3516-3519 **PERSPX (IIHX)** **m**

Perimeter between shell and shell side coolant in IHX.

3520-3523 **PERSPD (IDRX)** **m**

Perimeter between shell and shell side coolant in DRACS

Not implemented. See instead **ISGLNK** and **ICVNAK** and use **PERSPX**.

3524-3527 **PERPTX (IIHX)** **m**

Perimeter between shell side coolant and tube in IHX.

3528-3531 **PERPTD (IDRX)** m

Perimeter between shell side coolant and tube in DRACS.

Not implemented. See instead **ISGLNK** and **ICVNAK** and use **PERPTX**.

3532-3535 **PERTIX (IIHX)** m

Perimeter between tube and tube side coolant in IHX.

3536-3539 **PERTID (IDRX)** m

Perimeter between tube and tube side coolant in DRACS.

Not implemented. See instead **ISGLNK** and **ICVNAK** and use **PERTIX**.

3540-3543 **DSHIHX (IIHX)** m

Shell thickness in IHX.

3544-3547 **DSHDRC (IDRX)** m

Shell thickness in DRACS.

Not implemented. See instead **ISGLNK** and **ICVNAK** and use **DSHIHX**.

3548-3551 **DTUIHX (IIHX)** m

Tube thickness in IHX.

3552-3555 **DTUDRC (IDRX)** m

Tube thickness in DRACS.

Not implemented. See instead **ISGLNK** and **ICVNAK** and use **DTUIHX**.

3556-3559 **RCSHHX (IIHX)** J/m^3-K

Density * specific heat of shell in IHX.

3560-3563 **RCSHDR (IDRX)** J/m^3-K

Density * specific heat of shell in DRACS.

Not implemented. See instead **ISGLNK** and **ICVNAK** and use **RCSHHX**.

3564-3567 **RCTUHX (IIHX)** J/m^3-K

Density * specific heat of tube in IHX.

3568-3571 **RCTUDR (IDRX)** **J/m^3-K**

Density * specific heat of tube in DRACS.

Not implemented. See instead **ISGLNK** and **ICVNAK** and use **RCTUHX**.

3572-3575 **XKSHHX (IIHX)** **W/m-K**

Thermal conductance of shell in IHX.

3576-3579 **XKSHDR (IDRX)** **W/m-K**

Thermal conductance of shell in DRACS.

Not implemented. See instead **ISGLNK** and **ICVNAK** and use **XKSHHX**.

3580-3583 **XKTUHX (IIHX)** **W/m-K**

Thermal conductance of tube in IHX.

3584-3587 **XKTUDR (IDRX)** **W/m-K**

Thermal conductance of tube in DRACS.

Not implemented. See instead **ISGLNK** and **ICVNAK** and use **XKTUHX**.

3588-3591 **HFPIHX (IIHX)** **W/m^2-K**

Fouling heat-transfer coefficient for shell side flow in IHX.

If **HFPIHX** $< 1 \times 10^{-10}$ a default value of 1×10^{10} is applied.

3592-3595 **HFPDRC (IDRX)** **W/m^2-K**

Fouling heat-transfer coefficient for shell side flow in DRACS.

Not implemented. See instead **ISGLNK** and **ICVNAK** and use **HFPIHX**.

3596-3599 **HFIIHX (IIHX)** **W/m^2-K**

Fouling heat-transfer coefficient for tube side flow in IHX.

If **HFIIHX** $< 1 \times 10^{-10}$ a default value of 1×10^{10} is applied.

3600-3603 **HFIDRC (IDRX)** **W/m^2-K**

Fouling heat-transfer coefficient for tube side flow in DRACS.

Not implemented. See instead **ISGLNK** and **ICVNAK** and use **HFIIHX**.

3604-3607 **SLANTX (IIHX)**

Slant-height factor for IHX, tubeside.

3608-3611 **SLANTD (IDRX)**

Slant-height factor for DRACS, tubeside.

Not implemented. See instead **ISGLNK** and **ICVNAK** and use **SLANTX**.

3612-3649 **VOLGS0 (ICV)** **m^3**

Initial gas volume. Used only for **ITYPCV**(ICV) = 6,7,8,9,10.

3650 **C1PIPE**

Conductance coefficient for pipe.

Recommended value: 0.025.

3651 **C2PIPE**

Conductance coefficient for pipe.

Recommended value: 0.8.

3652 **C3PIPE**

Conductance coefficient for pipe.

Recommended value: 4.8.

3653 **C1IHX**

Conductance coefficient for HX, shell side. Also for tube side if both are same; if tube side different, enter tube side value in **C1IHXT** (heat exchanger).

Recommended value: 0.025.

3654 **C2IHX**

Conductance coefficient for HX, shell side. Also for tube side if both are same; if tube side different, enter tube side value in **C2IHXT** (heat exchanger).

Recommended value: 0.8.

3655 **C3IHX**

Conductance coefficient for HX, shell side. Also for tube side if both are same; if tube side different, enter tube side value in **C3IHXT** (heat exchanger).

Recommended value: 4.8.

3656 **C4IHX**

Conductance coefficient for HX, shell side. Also for tube side if both are same; if tube side different, enter tube side value in **C4IHXT** (heat exchanger).

Recommended value: 0.0.

3657-3719 **PMPHD (I,J,K)**

Homologous pump head coefficients for range J for pump type (specific speed) K. Used only if **IPMDFT**>0 and **IEMPMP**=2.

See *Homologous Pumps*.

Dimensions (7,3,3).

3720-3800 **PMPTQ (I,J,K)**

Homologous pump torque coefficients for range J for pump type K. Used only if **IPMDFT**>0 and **IEMPMP**=2.

See *Homologous Pumps*.

Dimensions (9,3,3).

AIR DUMP HEAT EXCHANGER

3801-3804 **RITB (IDHX)** **m**

Heat exchanger tube inner radius.

3805-3808 **ROTB (IDHX)** **m**

Tube outer radius.

3809-3812 **XKTB (IDHX)** **W/m-K**

Tube thermal conductivity.

3813-3816 **HOTB (IDHX)** **W/m^2-K**

Tube-to-air heat transfer coefficient.

3817-3820 **HITB (IDHX)** **W/m^2-K**

Tube-to-NaK or tube-to-Na heat transfer coefficient.

3821-3824 **AHT (IDHX)** **m^2**

Air side heat transfer area.

3825-3828 **SRLEN (IDHX)** **m**

Stack riser length.

3829-3832 **XKHXLS (IDHX)**

Loss coefficient, air side of heat exchanger.

3833-3836 **XKRLS (IDHX)**

Riser loss coefficient.

3837-3840 **XKSOLS (IDHX)**

Stack outlet loss coefficient.

3841-3844 **AHX (IDHX)** **m^2**

Air flow area at heat exchanger.

3845-3848 **AR (IDHX)** **m^2**

Riser cross sectional area.

3849-3852 **ASI (IDHX)** **m^2**

Stack inlet cross sectional area.

3853-3892 **XKSILS (K,IDHX)**

Table of stack inlet loss coefficient (`IFCDHX`=0) or air flow rate (`IFCDHX`=1) vs time.

3893-3932 **XKSITM (K,IDHX)** **s**

Times for `XKSILS` table.

3933 **AIRTMP** **K**

Air temperature.

FIRST STEAM GENERATOR

3934 **HFW** **J/kg**

Inlet feedwater enthalpy.

3935 **WEVI** **kg/s**

Inlet evaporator mass flow rate estimate (readjusted during steady-state calculations to yield correct sodium-side temperature drop).

3936 **PW** **Pa**

Evaporator waterside average pressure.

3937 **TNAX**

Not currently used.

3938 **XEVO**

Exit steam quality from the evaporator. Should be between 0 and 1 for **IEVAP** = 1 or 3.

GEOMETRIC CONSTANTS

3939 **EL** **m**

Evaporator active length.

3940 **DEW** **m**

Water hydraulic diameter per tube.

3941 **DNA** **m**

Tube outer diameter.

3942 **DENA** **m**

Sodium hydraulic diameter per tube.

3943 **ANA** **m^2**

Sodium flow area per tube.

3944 **POD**

Tube pitch to diameter ratio.

3945 **PINSW**

Number of tubes per unit.

EVAPORATOR HEAT TRANSFER NORMALIZATION

3946 **PWO** **Pa**

Evaporator average pressure.

3947 **TMO** **K**

Evaporator average metal temperature.

3948 **TNAO** **K**

Evaporator average sodium temperature.

3949 **HSCO** **J/kg**

H_2O: Average subcooled region enthalpy (estimated).

3950 **HFBO** **J/kg**

H_2O: Average post-DNB region enthalpy (estimated).

3951 **HSHO** **J/kg**

H_2O: Average superheat region enthalpy (estimated).

3952-3953 **GWO (2)** **kg/m^2-s**

H_2O: Average 2-phase mass velocity in the evaporator and superheater, respectively.

GEOMETRIC CONSTANTS

3954-3955 **QFACT (2)**

Heat transfer area multiplier for the evaporator and superheater respectively. For **IEVAP** = 1 or 3, **QFACT**(1) is internally adjusted such that the evaporator length and output steam quality correspond to their input values.

3956 **ELS** **m**

Superheater length.

INITIAL ENTHALPY

3957 **HWSHO** **J/kg**

Initial estimate of the superheat water exit enthalpy (internally changed during steady-state calculations).

INITIAL PRESSURE

3958 **PD** **Pa**

Initial drum pressure.

3959 **PHPTI** **Pa**

Initial high pressure turbine inlet pressure.

GEOMETRIC CONSTANTS

3960 **VOLD** **m^3**

Drum volume.

3961 **ELDRUM** **m**

Drum midpoint elevation measured from the evaporator inlet.

RECIRCULATION MOMENTUM

3962 **DPFCLD** **Pa**

Rated cold leg (drum to active inlet of evaporator) friction pressure loss.

3963 **CUTOFF**

Cutoff head fraction for recirculation pump rated head.

3964 **SLA** **1/m**

Recirculation loop characteristic length to flow area ratio.

3965 **DPFHOT** **Pa**

Rated hot leg (evaporator to drum) friction pressure loss.

3966 **REDPTO**

Rated evaporator water/steam Reynolds number.

3967 **FDPTO**

Evaporator friction pressure drop multiplier.

3968-3973 **FWTIME (6)** **s**

Array of time values corresponding to feedwater normalized flow rate profile (**FWPROF**).

3974-3979 **FWPROF (6)**

Feedwater flow at times given in **FWTIME**. Normalized to steady-state flow = 1.0. This option selected by **IFWC** = 2.

3980-3985 **PHTIME (6)** **s**

Array of time values corresponding to feedwater pump head normalized profile (PMPROF).

3986-3991 **PHPROF (6)**

Pump head profile at times given in **PHTIME**. Normalized to feedwater pump ΔP at steady-state. This option is turned on by setting **PHPROF**(1) non-zero.

3992-3993 **DUM**

Not currently used.

END OF FIRST STEAM GENERATOR INPUT SECOND STEAM GENERATOR

3994 **HFW** **J/kg**

Inlet feedwater enthalpy.

3995 **WEVI** **kg/s**

Inlet evaporator mass flow rate estimate (readjusted during steady-state calculations to yield correct sodium-side temperature drop).

3996 **PW** **Pa**

Evaporator waterside average pressure.

3997 **TNAX**

Not currently used.

3998 **XEVO**

Exit steam quality from the evaporator. Should be between 0 and 1 for **IEVAP** = 1 or 3.

GEOMETRIC CONSTANTS

3999 **EL** **m**

Evaporator active length.

4000 **DEW** **m**

Water hydraulic diameter per tube.

4001 **DNA** **m**

Tube outer diameter.

4002 **DENA** **m**

Sodium hydraulic diameter per tube.

4003 **ANA** **m^2**

Sodium flow area per tube.

4004 **POD**

Tube pitch to diameter ratio.

4005 **PINSW**

Number of tubes per unit.

EVAPORATOR HEAT TRANSFER NORMALIZATION

4006 **PWO** **Pa**

Evaporator average pressure.

4007 **TMO** **K**

Evaporator average metal temperature.

4008 **TNAO** **K**

Evaporator average sodium temperature.

4009 **HSCO** **J/kg**

H_2O: Average subcooled region enthalpy (estimated).

4010 **HFBO** **J/kg**

H_2O: Average post-DNB region enthalpy (estimated).

4011 **HSHO** **J/kg**

H_2O: Average superheat region enthalpy (estimated).

4012-4013 **GWO (2)** **kg/m^2-s**

H_2O: Average 2-phase mass velocity in the evaporator and superheater, respectively.

GEOMETRIC CONSTANTS

4014-4015 **QFACT (2)**

Heat transfer area multiplier for the evaporator and superheater respectively. For **IEVAP** = 1 or 3, **QFACT**(1) is internally adjusted such that the evaporator length and output steam quality correspond to their input values.

4016 **ELS** **m**

Superheater length.

INITIAL ENTHALPY

4017 **HWSHO** **J/kg**

Initial estimate of the superheat water exit enthalpy (internally changed during steady-state calculations).

INITIAL PRESSURE

4018 **PD** **Pa**

Initial drum pressure.

4019 **PHPTI** **Pa**

Initial high pressure turbine inlet pressure.

GEOMETRIC CONSTANTS

4020 **VOLD** **m^3**

Drum volume.

4021 **ELDRUM** **m**

Drum midpoint elevation measured from the evaporator inlet.

RECIRCULATION MOMENTUM

4022 **DPFCLD** **Pa**

Rated cold leg (drum to active inlet of evaporator) friction pressure loss.

4023 **CUTOFF**

Cutoff head fraction for recirculation pump rated head.

4024 **SLA** **1/m**

Recirculation loop characteristic length to flow area ratio.

4025 **DPFHOT** **Pa**

Rated hot leg (evaporator to drum) friction pressure loss.

4026 **REDPTO**

Rated evaporator water/steam Reynolds number.

4027 **FDPTO**

Evaporator friction pressure drop multiplier.

4028-4033 **FWTIME (6)** **s**

Array of time values corresponding to feedwater normalized flow rate profile (**FWPROF**).

4034-4039 **FWPROF (6)**

Feedwater flow at times given in **FWTIME**. Normalized to steady-state flow = 1.0. This option selected by **IFWC** = 2.

4040-4045 **PHTIME (6)** **s**

Array of time values corresponding to feedwater pump head normalized profile (PMPROF).

4046-4051 **PHPROF (6)**

Pump head profile at times given in **PHTIME**. Normalized to feedwater pump ΔP at steady-state. This option is turned on by setting **PHPROF**(1) non-zero.

4052-4053 **DUM**

Not currently used.

END OF SECOND STEAM GENERATOR INPUT THIRD STEAM GENERATOR

4054 **HFW** **J/kg**

Inlet feedwater enthalpy.

4055 **WEVI** **kg/s**

Inlet evaporator mass flow rate estimate (readjusted during steady-state calculations to yield correct sodium-side temperature drop).

4056 **PW** **Pa**

Evaporator waterside average pressure.

4057 **TNAX**

Not currently used.

4058 **XEVO**

Exit steam quality from the evaporator. Should be between 0 and 1 for **IEVAP** = 1 or 3.

GEOMETRIC CONSTANTS

4059 **EL** **m**

Evaporator active length.

4060 **DEW** **m**

Water hydraulic diameter per tube.

4061 **DNA** **m**

Tube outer diameter.

4062 **DENA** **m**

Sodium hydraulic diameter per tube.

4063 **ANA** **m^2**

Sodium flow area per tube.

4064 **POD**

Tube pitch to diameter ratio.

4065 **PINSW**

Number of tubes per unit.

EVAPORATOR HEAT TRANSFER NORMALIZATION

4066 **PWO** **Pa**

Evaporator average pressure.

4067 **TMO** **K**

Evaporator average metal temperature.

4068 **TNAO** **K**

Evaporator average sodium temperature.

4069 **HSCO** **J/kg**

H_2O: Average subcooled region enthalpy (estimated).

4070 **HFBO** **J/kg**

H_2O: Average post-DNB region enthalpy (estimated).

4071 **HSHO** **J/kg**

H_2O: Average superheat region enthalpy (estimated).

4072-4073 **GWO (2)** **kg/m^2-s**

H_2O: Average 2-phase mass velocity in the evaporator and superheater, respectively.

GEOMETRIC CONSTANTS

4074-4075 **QFACT (2)**

Heat transfer area multiplier for the evaporator and superheater respectively. For **IEVAP** = 1 or 3, **QFACT**(1) is internally adjusted such that the evaporator length and output steam quality correspond to their input values.

4076 **ELS** **m**

Superheater length.

INITIAL ENTHALPY

4077 **HWSHO** **J/kg**

Initial estimate of the superheat water exit enthalpy (internally changed during steady-state calculations).

INITIAL PRESSURE

4078 **PD** **Pa**

Initial drum pressure.

4079 **PHPTI** **Pa**

Initial high pressure turbine inlet pressure.

GEOMETRIC CONSTANTS

4080 **VOLD** **m^3**

Drum volume.

4081 **ELDRUM** **m**

Drum midpoint elevation measured from the evaporator inlet.

RECIRCULATION MOMENTUM

4082 **DPFCLD** **Pa**

Rated cold leg (drum to active inlet of evaporator) friction pressure loss.

4083 **CUTOFF**

Cutoff head fraction for recirculation pump rated head.

4084 **SLA** **1/m**

Recirculation loop characteristic length to flow area ratio.

4085 **DPFHOT** **Pa**

Rated hot leg (evaporator to drum) friction pressure loss.

4086 **REDPTO**

Rated evaporator water/steam Reynolds number.

4087 **FDPTO**

Evaporator friction pressure drop multiplier.

4088-4093 **FWTIME (6)** **s**

Array of time values corresponding to feedwater normalized flow rate profile (**FWPROF**).

4094-4099 **FWPROF (6)**

Feedwater flow at times given in **FWTIME**. Normalized to steady-state flow = 1.0. This option selected by **IFWC** = 2.

4100-4105 **PHTIME (6)** **s**

Array of time values corresponding to feedwater pump head normalized profile (PMPROF).

4106-4111 **PHPROF (6)**

Pump head profile at times given in **PHTIME**. Normalized to feedwater pump ΔP at steady-state. This option is turned on by setting **PHPROF**(1) non-zero.

4112-4113 **DUM**

Not currently used.

END OF THIRD STEAM GENERATOR INPUT FOURTH STEAM GENERATOR

4114 **HFW** **J/kg**

Inlet feedwater enthalpy.

4115 **WEVI** **kg/s**

Inlet evaporator mass flow rate estimate (readjusted during steady-state calculations to yield correct sodium-side temperature drop).

4116 **PW** **Pa**

Evaporator waterside average pressure.

4117 **TNAX**

Not currently used.

4118 **XEVO**

Exit steam quality from the evaporator. Should be between 0 and 1 for **IEVAP** = 1 or 3.

GEOMETRIC CONSTANTS

4119 **EL** **m**

Evaporator active length.

4120 **DEW** **m**

Water hydraulic diameter per tube.

4121 **DNA** **m**

Tube outer diameter.

4122 **DENA** **m**

Sodium hydraulic diameter per tube.

4123 **ANA** **m^2**

Sodium flow area per tube.

4124 **POD**

Tube pitch to diameter ratio.

4125 **PINSW**

Number of tubes per unit.

EVAPORATOR HEAT TRANSFER NORMALIZATION

4126 **PWO** **Pa**

Evaporator average pressure.

4127 **TMO** **K**

Evaporator average metal temperature.

4128 **TNAO** **K**

Evaporator average sodium temperature.

4129 **HSCO** **J/kg**

H_2O: Average subcooled region enthalpy (estimated).

4130 **HFBO** **J/kg**

H_2O: Average post-DNB region enthalpy (estimated).

4131 **HSHO** **J/kg**

H_2O: Average superheat region enthalpy (estimated).

4132-4133 **GWO (2)** **kg/m^2-s**

H_2O: Average 2-phase mass velocity in the evaporator and superheater, respectively.

GEOMETRIC CONSTANTS

4134-4135 **QFACT (2)**

Heat transfer area multiplier for the evaporator and superheater respectively. For **IEVAP** = 1 or 3, **QFACT**(1) is internally adjusted such that the evaporator length and output steam quality correspond to their input values.

4136 **ELS** **m**

Superheater length.

INITIAL ENTHALPY

4137 **HWSHO** **J/kg**

Initial estimate of the superheat water exit enthalpy (internally changed during steady-state calculations).

INITIAL PRESSURE

4138 **PD** **Pa**

Initial drum pressure.

4139 **PHPTI** **Pa**

Initial high pressure turbine inlet pressure.

GEOMETRIC CONSTANTS

4140 **VOLD** **m^3**

Drum volume.

4141 **ELDRUM** **m**

Drum midpoint elevation measured from the evaporator inlet.

RECIRCULATION MOMENTUM

4142 **DPFCLD** **Pa**

Rated cold leg (drum to active inlet of evaporator) friction pressure loss.

4143 **CUTOFF**

Cutoff head fraction for recirculation pump rated head.

4144 **SLA** **1/m**

Recirculation loop characteristic length to flow area ratio.

4145 **DPFHOT** **Pa**

Rated hot leg (evaporator to drum) friction pressure loss.

4146 **REDPTO**

Rated evaporator water/steam Reynolds number.

4147 **FDPTO**

Evaporator friction pressure drop multiplier.

4148-4153 **FWTIME (6)** **s**

Array of time values corresponding to feedwater normalized flow rate profile (**FWPROF**).

4154-4159 **FWPROF (6)**

Feedwater flow at times given in **FWTIME**. Normalized to steady-state flow = 1.0. This option selected by **IFWC** = 2.

4160-4165 **PHTIME (6)** **s**

Array of time values corresponding to feedwater pump head normalized profile (PMPROF).

4166-4171 **PHPROF (6)**

Pump head profile at times given in **PHTIME**. Normalized to feedwater pump ΔP at steady-state. This option is turned on by setting **PHPROF**(1) non-zero.

4172-4173 **DUM**

Not currently used.

END OF FOURTH STEAM GENERATOR INPUT

4174-4183 **VSLEXP (K)** **m/K**

Length times thermal expansion coefficient for the K element or compressible volume. Used to represent the vessel wall for the control rod drive thermal expansion feedback calculation.

COMPONENT-TO-COMPONENT HEAT TRANSFER

4184-4215 **HABYBY (K,IBYP)** **W/m-K**

Heat transfer coefficient x area per unit height for bypass channel-bypass channel heat transfer. Dimensions (4,4). See **IBYBY**.

4216-4245 **HAELHT (K)** **W/m-K or W/K**

Heat transfer coefficient x area per unit length (W/m-K) for heat transfer from wall of liquid element **IELHT**(K) to liquid in compressible volume **IELHT2**(K). For heat transfer from compressible volume to compressible volume, **HAELHT** is h x area (W/K).

RVACS

4246-4251 **XLRVC (IRVC)** **m**

Length of RVACS section. Used only if section K is a compressible volume.

4252-4257 **PERVAC (IRVC)** **m**

RV perimeter.

TABLE RVACS

4258-4269 **RVHTAB (K)**

Table of h vs T. See **IRVOPT** > 0.

4270-4281 **RVHTMP (K)**

T for **RVHTAB** table.

DETAILED RVACS

4258-4263 **EPSRV (IRVC)**

Emissivity of reactor vessel.

4264-4269 **EPSGV (IRVC)**

Emissivity of guard vessel inner surface.

4270-4275 **PERGV (IRVC)**

Perimeter, GV-air.

4276-4281 **PERFS (IRVC)**

Perimeter, finned shell inner surface - air.

TIME STEP CONTROLS

4282 **EPSF**

Maximum fractional change in a liquid segment flow rate per PRIMAR time step.

4283 **EPSFC**

Maximum fractional change in core inlet flow rate per PRIMAR time step.

STEADY-STATE INITIALIZATION

4284-4287 **DTIHX (IIHX)**

Steady state temperature drop across IHX (primary side). Used only if `ISSIHX` > 0.

4288-4299 **DHPMP (IPMP)**

Steady state pump head. Used only if ISSPMP > 0.

BYPASS CHANNEL HEAT TRANSFER CORRELATIONS

4300 **C1BY2**

Second coefficient set for bypass channel heat transfer coefficient.

4301 **C2BY2**

See `C1BY2`.

4302 **C3BY2**

See `C1BY2`.

4303 **C1BY3**

Third coefficient set for bypass channel heat transfer coefficient.

4304 **C2BY3**

See **C1BY3**.

4305 **C3BY3**

See **C1BY3**.

4306 **C1BY4**

Fourth coefficient set for bypass channel heat transfer coefficient. See **IHTBYB**, **IHTBYD**, and **C1BY**.

4307 **C2BY4**

See **C1BY4**.

4308 **C3BY4**

See **C1BY4**.

IHX HEAT TRANSFER CORRELATIONS

See **C1IHX**, **C2IHX**, and **C3IHX**.

4309 **C1IHXT**

Heat transfer correlation coefficients for the tube side of an HX.

4310 **C2IHXT**

Heat transfer correlation coefficients for the tube side of an HX.

4311 **C3IHXT**

Heat transfer correlation coefficients for the tube side of an HX.

4312 **C4IHXT**

Heat transfer correlation coefficients for the tube side of an HX.

ANNULAR ELEMENTS

4313-4342 **WALMC2 (IAEL)** **J/m-K**

Second wall mass x heat capacity / length for annular element IAEL.

4343-4372 **WALLH2 (IAEL)** **W/m^2-K**

Second wall heat transfer coefficient.

4373-4402 **PERWL2 (IAEL)** **m**

Second wall perimeter.

RVACS, CONTINUED

4403 **TAIRVC** **K**

Air inlet temperature. When using the detailed RVACS model, the air property correlations are valid between 255.4 K and 1088.7 K.

Ignored if `IDRVACSTin` > 0.

4404 **ZBRVC** **m**

Z at bottom of RVACS.

THE FOLLOWING RVACS INPUT IS ONLY FOR THE DETAILED MODEL

4405 **C1RV**

Heat transfer correlation coefficient for air in the RVACS. See Eq. (5.4-138).

4406 **C2RV**

Heat transfer correlation coefficient for air in the RVACS.

4407 **C3RV**

Heat transfer correlation coefficient for air in the RVACS.

See also `C4RV`.

4408 **XLAIRV** **m**

Length of air inlet section.

4409 **DHAIRV** **m**

Dh, air inlet section.

4410 **AARIRV** **m^2**

Area, air inlet section.

4411 **XLAORV** **m**

Length of air outlet (stack).

4412 **DHAORV** **m**

Dh, stack.

4413 **AARORV** **m^2**

Area, stack

4414-4419 **AIRARV (IRVC)** **m^2**

Air flow area, GV-FS.

4420-4425 **AIRAR2 (IRVC)** **m^2**

Air flow area, FS - concrete wall.

4426-4431 **PERGVO (IRVC)** **m**

Perimeter for radiation from guard vessel outer surface to outer shell inner surface.

4432-4437 **PERFSI (IRVC)**

Not currently used.

4438-4443 **PERFSO (IRVC)** **m**

Perimeter for radiation from finned shell outer surface to outer wall, also finned shell outer surface to air.

4444-4449 **HGASRV (IRVC)** **W/m^2-K**

Gas h, RV - GV.

4450-4455 **SLRVC (IRVC)**

Slope:

= 0, Vertical.
= 1, Horizontal.

4456-4461 **HFSRV (IRVC)** **W/m^2-K**

Heat transfer coefficient across finned shell.

4462-4467 **HCONRV (IRVC)** **W/m^2-K**

Heat transfer coefficient between concrete nodes.

COMPONENT - COMPONENT HEAT TRANSFER

4468-4497 **CPCPMP (K)**

Multiplicity for component - component heat transfer, default = 1.0.

RVACS

4498-4503 **GVMC (IRVC)** **J/m-K**

Mass x specific heat / length, guard vessel.

4504-4509 **FSMCI (IRVC)** **J/m-K**

MC/L, finned shell inner node.

4510-4515 **FSMCO (IRVC)** **J/m-K**

MC/L, FS outer node.

4516-4521 **CRMCI (IRVC)** **J/m-K**

MC/L, concrete inner node.

4522-4527 **CRMCO (IRVC)** **J/m-K**

MC/L, concrete outer node.

4528-4533 **TW6RV (IRVC)** **K**

Temperature of outer wall node.

4534 **EPSFS**

Emissivity, finned shell outer surface.

4535 **SIGSTB** **W/m^2-K**

Stefan-Boltzmann constant. Suggested value: 5.672E-8.

4536 **RW5RV** **m^2-K/W**

Thermal resistance between surface of outer wall and inner node (node 5).

4537 **REYTRV**

Reynolds number for transition from laminar to turbulent in air. Overwritten by the code. See Eq. (5.4-123).

4538 **AFRTRV**

Turbulent friction factor = **AFRTRV***Re****BFRTRV**.

4539 **AFRLRV**

Laminar friction factor = **AFRLRV**/Re.

4540 **ORFIN**

Inlet orifice coefficient.

Ignored if **IDRVACSKin** > 0.

4541 **XLUNRV** m

Length of upper node between liquid level and top of vessel.

4542 **BFRTRV**

Turbulent friction factor parameter, see **AFRTRV**.

4543 **UNGVMC** J/m-K

MC/L for GV in upper node.

4544 **UNFSMC** J/m-K

MC/L for FS inner node in upper node.

4545 **AIRAUN** m^2

Air flow area, upper node.

4546 **EPSOW**

Emissivity, outer wall.

4547-4552 **EPSGVO (IRVC)**

Emissivity, guard vessel outer surface.

4553-4558 **EPSFSI (IRVC)**

Emissivity, finned shell inner surface.

CHECK VALVE TABLES

4559-4618 **CKVORF (K,ICKV)**

Check valve coefficient table. Dimensions: 10,6.

4619-4678 **CKVFLO (K,ICKV)**

Check valve normalized flow. Dimension: 10,6.

DEGREE OF IMPLICITNESS

4679 **THT2CV**

Minimum degree of implicitness for a compressible volume for liquid flow rate calculations.

Suggested values: 0.5 for rapid transients, 0.75 for slow transients

PUMP CAVITATION

4680 **CPC0**

Pump cavitation coefficients required net positive suction head =

S*S(**CPC0**+ **CPC1***Z+**CPC2***Z^2+**CPC3***Z^3+**CPC4***Z^4).

Z = W/S

W = normalized flow

S = normalized pump speed

Typical values for FFTF:

CPC0 = 457,488
CPC1 = -2,161,072
CPC2 = 4,266,337
CPC3 = -3,636,676
CPC4 = 1,141,003

4681 **CPC1**

See **CPC0**.

4682 **CPC2**

See **CPC0**.

4683 **CPC3**

See **CPC0**.

4684 **CPC4**

See CPC0.

MULTIPLE INLET/OUTLET PLENA

4685-4722 **TPLCV (ICV)** **K**

Temperature of volume ICV. Entered only for inlet plena to core or bypass channels. Not required for IFMIOP = 0.

4723-4760 **PPLCV (ICV)** **Pa**

Pressure of volume ICV. Entered only for outlet plena to core or bypass channels. Not required for IFMIOP = 0.

4761-4798 **ZPLENC (ICV)** **m**

Reference elevation for volume ICV. Entered only for inlet and outlet plena for core and bypass channels. (ZPLENL and ZPLENU are set from ZPLENC in the channel subroutines). Not required for IFMIOP = 0.

ELEMENT/WALL THERMAL ADJUSTMENT

4799-4938 **WALTHK (IELL)** **m**

Element wall thickness, used only if ITHPEN > 0.

4939-4976 **THKWAL (ICV)** **m**

Compressible volume wall thickness, used only if ITHPEN > 0.

4977 **TAUPEN** **s**

Time constant for thermal penetration depth adjustments to wall thermal treatments, used only if ITHPEN > 0.

4978-5007 **WALTH2 (IAEL)** **m**

Annular element second wall thickness, used only if ITHPEN > 0.

STRATIFIED COMPRESSIBLE VOLUME

5008 **RCORE** **m**

Core radius for use in the Richardson number.

5009-5017 **HCSTWL (IW)** **W/m^2-K**

Coolant heat transfer coefficient at the inner surface of the wall section.

5018-5026 **HCSTW2 (IW)** W/m^2-K

Coolant heat transfer coefficient at the outer surface of the wall section.

5027-5035 **ASTWL (IW)** m^2

Area of the wall section.

5036-5107 **HINVWL (I,IW)** m^2-K/W

Thickness/thermal conductivity of node I in the wall section. I = 1 - 8.

5108-5179 **XMCSTW (I,IW)** J/K

Mass x heat capacity of node I in the wall.

5180-5182 **ZINST (ICVST)** m

Z of inlet, used only for a vertical wall. Otherwise **ZOUTEL**(IELL) is used.

5183-5185 **VOLBLI (ICVST)** m^3

Volume below the inlet, not used if there is a cover gas in the CV. In this case, the information is obtained from other input for the CV.

5186-5188 **EPSTST (ICVST)** K

Minimum temperature difference for switching stages.

5189-5191 **XLENTR (ICVST)** m

Entrainment length. A hot plume with a flow rate WH, rising through a cool layer of thickness DZ, will entrain cool liquid at a rate (DZ/**XLENTR**) * WH.

PARAMETRIC TWO-REGION IHX MODEL

5192-5195 **DTSIHX (IIHX)** K

Initial temperature difference between regions in parametric two-region IHX model for **IHXCLC**(IIHX) < 0 and **DTMPTB**(1) = 1.0.

5196-5199 **VSIHX1 (IIHX)** m^3

First region volume in parametric two-region IHX model for **IHXCLC**(IIHX) < 0 and **DTMPTB**(1) = 1.0.

5200-5203 **VSIHX2 (IIHX)** m^3

Second region volume in parametric two-region IHX model for **IHXCLC**(IIHX) < 0 and **DTMPTB**(1) = 1.0.

THICK-WALLED PIPES

5204-5243 **TPWMC (L,K)** **J/m-K**

Mass*heat capacity/length for wall node L+1 of thick-walled pipe K. Dimensions (4,10). Note: Use **WALLMC**(IELL) for node 1.

5244-5283 **TPWH (L,K)** **W/m^2-K**

Thermal conductivity/node thickness for node L+1 of thick walled pipe K. Dimensions (4,10). Note: Use **WALLH**(IELL) for node 1 for h from the mid-point of the first wall node to the wall surface.

STEADY-STATE INITIALIZATION

5284-5293 **TCVSSI (II)** **K**

Steady-state coolant temperature in compressible volume **ICVSSI**(II).

5294-5303 **PCVSSI (II)** **Pa**

Steady-state liquid pressure in compressible volume **ICVSSI**(II).

Detailed Air Dump Heat Exchanger Model

IADHX(IDHX) > 0

Not currently used.

5304-5307 **EFFIN (IDHX)** **m**

Fin height. Not currently used

5308-5311 **FINTHK (IDHX)** **m**

Fin thickness. Not currently used.

5312-5315 **SPFIN (IDHX)** **m**

Center-to-center spacing between fins. Not currently used.

5316-5319 **XLTUBE (IDHX)** **m**

Tube spacing parallel to flow. Not currently used.

5320-5323 **XTTUBE (IDHX)** **m**

Tube spacing perpendicular to the flow. Not currently used.

5324 **FINVNT** **-**

Inverse scaling factor for the detailed RVACS thermal inertia during the null transient simulation. Default: 10.0

5325 **C4RV** -

Heat transfer correlation coefficient for air in the RVACS. See Eq. (5.4-138).

See also **C1RV**, **C2RV**, and **C3RV**.

5326 **SPLITF** -

Splitting factor for the air flow into the lower cavity mixing region. See Eq. (5.4-115).

0.0 <= **SPLITF** <= 1.0

Only relevant if **ILGVHT** = 1.

5327 **HTCLGV** W/m^2-K

Air heat transfer coefficient between the lower guard vessel and the air in the lower cavity mixing region. See Eq. (5.4-110).

Only relevant if **ILGVHT** = 1.

5328-5500 **DMP4IN**

Not currently used.

2.7.11 Block 51 — INPCHN — Channel-Dependent Options and Integer Input

1 **IDBUGV**

Channel-dependent debug flag. (See **IDBUG0** and **IERSTP**).

= 0, For no debugging output.
= 1, For steady-state debugging output.

Time step control output for 2 ≤ **IDBUG0** ≤ 6.

= 2, Time step, also steady-state fuel behavior output if **IDBUG0** = 0.
= 3, Coolant calculation output. (See **IERSTP**)
= 4, For more coolant debugging output. (See **IERSTP**)
= 5, For transient heat-transfer debugging output
= 6, Print out all reactivity changes
= 7, Primary loop calculation debug
= 8, Coolant-cladding temperature calculations only

2 IERSTP

Main time step number when debugging output starts. Relevant only if **IDBUGV** > 0. If **IERSTP** is negative, debug output starts on step |**IERSTP**| of a null transient (**ISSNUL** > 0).

3 IRHOK

Fuel density, heat capacity and thermal conductivity selection parameter.

≤ 0, Temperature dependent tabular forms. See **RHOTAB**, **XKTAB**, and **CPFTAB** for tables.
> 0, Correlations that depend on temperature, and in case of binary/ ternary fuel, also on composition.

Correlations for oxide fuel (IMETAL=0):

> 0, Temperature and porosity dependent correlations for fuel density. Inputs **COEFDS**, **COEFDL** are required.
= 1, Correlation for fuel conductivity from GEAP-13582.
= 2, Correlation for fuel conductivity from WARD-FTM-FI-RJS-17. Input **COEFK** is required.
= 3, Correlation for fuel conductivity based on COMETH3J-FBR.

Note: The parameter values associated with **IRHOK**=1 or 3 need not be input. For oxide fuel, tabular heat capacity is used even if **IRHOK** > 0.

Correlations for U-Fs fuel (IMETAL=1):

> 0, Correlation for fuel density uses input **REFDEN** with thermal expansion function subprogram ALPHF. Correlation for fuel conductivity depends on temperature. Porosity and sodium logging is accounted for by the method of ANL/RAS 85-19.

Note: For U-Fs fuel, tabular heat capacity is used even if **IRHOK** > 0.

Correlations for ternary/binary fuel (IMETAL=2,3):

Note: Input **IFUELM** is required for this fuel.

IFUELM:

= 0, IFR handbook data interpolations;
= 1, ANL/RAS 85-19 correlations;
= 2, Mark-V fuel or U-10Zr fuel according to input **IMETAL** = 2 or 3.

For **IFUELM**=0:

> 0, Correlation for fuel density depends on temperature. Input **RHOZN** is required. Correlation for fuel heat capacity depends on alloy composition/type and temperature. Inputs

PUZRTP , **TFSOL** , **TFLIQ** are required. Correlation for fuel conductivity depends on composition/ type, temperature, porosity and sodium logging. Inputs **PRSTY** and **XLOGNA** are required.

For **IFUELM**=1:

> 0, Correlation for fuel density depends on alloy composition and temperature. Correlation for fuel heat capacity depends on alloy composition and temperature. Correlation for fuel conductivity depends on composition, temperature, porosity and sodium logging. Inputs **POROSS** , **PORMSS** , **PORCSS** , **FPORNA** are required.

For **IFUELM**=2:

> 0, Correlation for Mark-V & U-10Zr fuel densities. Input composition from array PUZRTP is used with thermal expansion. Correlation for Mark-V & U-10Zr fuel heat capacities depend on temperature. Correlation for Mark-V & U-10Zr fuel conductivities depend on composition, temperature and burnup. Input **BURNFU** is required, based on memos of Billone to Briggs dated 1/14/91, 3/8/91 & 10/21/91.

4 NPLN

Number of segments in gas plenum.

$1 \leq$ **NPLN** ≤ 6.

5 NREFB

Number of reflector zones below the pin section.

$1 \leq$ **NREFB** ≤ 5.

6 NREFT

Number of reflector zones above the pin section.

$1 \leq$ **NREFT** ≤ 5.

NREFB + **NREFT** ≤ 6.

7-13 NZNODE (KZ)

Number of segments in zone KZ. (KZ ≤ 7).

Total number of all segments in all zones ≤ 48. Only one segment per node is necessary, but if the LEVITATE or PLUTO2 region can extend into a node the segments there should be in the range from 0.03 meters to 0.1 meters. Neighboring cells for PLUTO2 and LEVITATE should not differ much in length. A length ratio of 1.5 is still reasonable.

14 NT

Number of radial temperature nodes within the fuel,

$3 <$ **NT** < 12.

15 **IFUELV**

Table number of property value to be used for core fuel.

0 < **IFUELV** ≤ **IFUEL1**.

16 **IFUELB**

Table number of property value to be used for blanket fuel. 0 < **IFUELB** ≤ **IFUEL1**.

17 **ICLADV**

Table number of property value to be used for cladding table.

18 **NGRDSP**

Number of spacer grids in pin. 2 ≤ **NGRDSP** ≤ 10.

= 0, No spacer grids.

19 **KTING**

Fission-gas release model option.

= 0, Isotropic release model.
= 1, Weisman model.

20 **NAXOP**

Model selection for axial expansion and crack volume.

NAXOP	Crack Volume	Axial Plane Strain	Axial Swelling
0	No	No	No
1	No	Yes	No
2	No	No	Yes
3	No	Yes	Yes
4	Yes	No	No
5	Yes	Yes	No
6	Yes	No	Yes
7	Yes	Yes	Yes

If **NAXOP** = X with X above, mixed plane strain.

= 3X with X above, cladding controlled axial plane strain.
= 2X with X above, then free fuel axial plane strain.

= 1X constrained plane strain.

21 MSTEP

Pre-transient constant power and power-change specifications.

≤ 8, Number of power intervals defined by **PLIN** and **TPLIN**. One required for each power change. One required for each constant power interval.
> 8, Channel-dependent *Table ID* defining normalized power and flow per pin during pre-transient characterization.

If the channel-dependent table does not exist, a global table must exist with the corresponding ID. For **MSTEP** > 8, see the *example input* in Section 9.6.

22 ITAU

Irradiation induced SS316 cladding, **ICTYPE** = 1, swelling incubation parameter options.

=-2, No cladding swelling.
=-1, Lower limit.
= 0, Nominal value.
> 0, Upper limit.

23 IRATE

Irradiation induced cladding swelling rate option. Applicable to SS316 (**ICTYPE** = 1), D9 (**ICTYPE** = 2), and HT9 (**ICTYPE** = 3).

=-2, No cladding swelling.

For SS316:

=-1, Lower limit.
= 0, Nominal value.
> 0, Upper limit.

For D9:

= 0, Swelling rate based on B. J. Makenas.
= 1, Swelling rate based on Metallic Fuel Handbook..

For HT9:

= 0, Swelling rate based on D. L. Porter.
= 1, Swelling rate based on Metallic Fuel Handbook..

24 **IHGAP**

Fuel-cladding gap conductance selection.

= 0, SAS3D method for calculating HB (See **AHBPAR**, **BHBPAR**, **CHBPAR**, **HBMAX**, **HBMIN**, and **HBPAR**).
= 1, SAS4A method (Ross-Stoute model).

Note: When using the simple bond gap conduction model (**ISSFU2** = -1), set **IHGAP** =0.

25 **NPIN**

Number of pins per subassembly.

26 **NSUBAS**

Number of subassemblies in channel.

27 **MZUB**

Number of segments in upper blanket.

28 **MZLB**

Number of segments in lower blanket.

29 **IHEX**

= 0, No hexadecimal printout of the sum of the coolant temperatures.
> 0, Hexadecimal printout of the sum of the coolant temperatures.

30 **IRELAX**

Stress relaxation options. Use **IRELAX** = 0.

= 0, No relaxation.
= 1, Creep law estimate, not operational.
= 2, Exact solution, not operational.

31 NGRAIN

Model selection for grain growth theory.

= 0, Limited grain growth theory.
> 0, Unlimited grain growth theory, where **NGRAIN** is the grain diameter exponent in Eq. (8.3-9).

Suggested value: 4.

32 ISSFUE

= 0, Bypass dynamical model calculations in steady-state, no fuel restructuring and no deformation of fuel and cladding.
= 1, Use dynamic calculation of DEFORM in steady-state.

33 IRAD

Not currently used.

34 ILAG

= 0, Use Eulerian coolant temperature calculation until flow reversal.
= 1, Use Lagrangian coolant temperature calculation from the start.

Suggested value: 0.

35 NOSTRN

Option to avoid radial strain in the cladding even if conditions would produce strain.

= 0, Strain cladding if conditions warrant.
= 1, Allow no cladding strain.

36 JRUPT

Not currently used.

37-44 NPLIN (M)

Number of subdivisions in each of the **MSTEP** divisions. There should be enough subdivisions to allow for reasonable feedback of the restructuring into the thermal calculation. At least 4 subdivisions for each power change and enough constant power subdivisions so that each does not exceed about 10 days (preferably 2 days during early irradiation).

(M = 1, **MSTEP**)

45 **IROR**

Controls assumption used when molten cavity extends to the cracked fuel zone.

= 0, Cavity pressure reduced by R/R before acting on cladding, no crack volume included in cavity.
= 1, Cavity pressure acts on the cladding, all crack volume included in cavity if molten to cracked region.

46 **JPRNT1**

The lowest axial node for which debug output is produced from DEFORM.

47 **JPRNT2**

The highest axial node for which debug output is produced from DEFORM.

Note: The debug output is produced for all nodes from **JPRNT1** to **JPRNT2** inclusive. If these two values are set to 0 but the time step controls and **IDBUGF** are activated, then the debug output is from the molten cavity routine and the axial expansion and feedback calculation only (the axial node independent part of the calculation).

48 **NNBUG1**

The time step at which to start the debug output from DEFORM.

49 **NNBUG2**

The time step which is the last time step for debug output from DEFORM

50 **IDBUGF**

The control for the type of debug output desired from DEFORM.

= 3, **NNBUG1** and **NNBUG2** refer to the steady-state time step.
=13, **NNBUG1** and **NNBUG2** refer to the transient time step.
=23, **NNBUG1** and **NNBUG2** refer to both the steady-state and the transient time steps.
=-1, Initiates a print option that writes out when DEFORM (the transient DEFORM driver) is entered and left beginning at transient time step **NNBUG1**.
=-2, This is a special option that produces standard DEFORM output at every time step from **NNBUG1** to **NNBUG2** inclusive. **NNBUG1** and **NNBUG2** must be entered as negative numbers. This is active only in the transient calculation.

51-58 **NSKIP (M)**

Print control for each of the **MSTEP** divisions in DEFORM. There is one value for each **NPLIN** value.

< NPLIN, the DEFORM results are printed after each NSKIP subdivisions.
= NPLIN, the DEFORM results are printed at the end of the MSTEP division.
> NPLIN, no DEFORM results are printed.

(M = 1, MSTEP)

59 **MPL1**

Plots channel pressure history at node MPL1. The nodes MPL1 - MPL7 must be a node in the zone KZPIN.

60 **MPL2**

Plots channel pressure history at node MPL2.

61 **MPL3**

Plots channel pressure history at node MPL3.

62 **MPL4**

Plots channel temperature history at node MPL4.

63 **MPL5**

Plots channel temperature history at node MPL5.

64 **MPL6**

Plots channel temperature history at node MPL6.

65 **MPL7**

Plots channel coolant volumetric flow rate at node MPL7. (MPL7 should correspond to the location of upper flowmeter in experiments, the volumetric flowrate of lower coolant slug is plotted with it.)

66 **MPL8**

Not currently used.

67 **MPL9**

Not currently used.

68 **KKSBTP**

Not currently used.

69 **KKSBRI**

Not currently used.

70 **NRPI**

Not currently used.

71 **NRPI1**

Number of pins per assembly that are assumed to fail (number of unfailed pins is **NPIN** - **NRPI1**) when the fuel-pin failure option **MFAIL** is satisfied.

72 **NRPI2**

Not currently used.

73 **NRPI3**

Not currently used.

74 **IPSIZE**

Fuel particle size option in PLUTO2 and LEVITATE.

= 1, **RAFPLA** is used from the initial fuel ejection to time **TIFP**.
= 2, **RAFPSM** is used all the way.

75 **IBUGPL**

Debug flag, should currently always be 0.

76 **ICFINE**

= 0, Automatic time step selection in LEVITATE and PLUTO2 using **DTPLIN** as the initial and later on the minimum time step.
= 1, Initial time step **DTPLIN** is used all the time if the main and primary loop time steps remain multiples of **DTPLIN**. If this is not the case, then the PLUTO2 and LEVITATE time steps are temporarily smaller than **DTPLIN**.

77 **IPRINT**

Should currently always be 0.

78 **IPLOT**

> 0, Plotting file (Logical units 12, 13 and and 14) for PLUTO2 is written every **IPLOT** milliseconds if **IPLOT** is greater than zero.

= 0, No plotting data is saved.

79 **IBGO**

Debug printout starting at LEVITATE and PLUTO2 cycle **IBGO**.

80 **IBSTOP**

PLUTO2 or LEVITATE printout debug ends at cycle **IBSTOP**.

81 **IBNEW**

Debug levels:

= 0, No debug output.
= 1, Time step debug.
= 2, Some debug output.
= 3, More debug output.
= 4, Lots of debug output.

82 **IPGO**

Between cycles **IPGO** and **IPSTOP** regular full LEVITATE or PLUTO2 output at every **IPNEW** cycles.

83 **IPSTOP**

See **IPGO**.

84 **IPNEW**

See **IPGO**.

85 **ICLADB**

= 0, Cladding is ignored after melting.
= 1, Cladding motion occurs.
= 2, Cladding motion does not occur, but heat transfer to molten cladding does occur.

86 **MFAIL**

Fuel-pin failure option. (See **FSPEC**).

= 0, No pin failure.
= 1, **FSPEC** is failure time.
= 2, **FSPEC** is fuel failure temperature.
= 3, **FSPEC** is fuel mass melt fraction at failure.

= 4, **FSPEC** is cavity pressure at failure.

= 5, **FSPEC** is cladding yield stress at failure. Not yet operational.

= 6, Eutectic penetration correlation for U-5fs fuel. **FSPEC** is eutectic temperature.

= 7, Failure criteria consistent with PLUTO2/LEVITATE rip propagation model. Functional ultimate tensile strength used and fully cracked fuel assumed.

= 8, Eutectic penetration and stress-based failure for metal fuel.

= 9, Failure time based on melt fraction at **FSPEC** with location based on maximum **MFAIL** = 7 failure criterion.

87 **IFAIL**

Relevant for **MFAIL** = 2, the radial node to test.

88 **JFAIL**

Relevant for **MFAIL** = 1,2,3,4,5, the axial fuel-pin node to test.

Note: If **IFAIL** or **JFAIL** are not specified, the peak value of the pertinent failure quantity is used. If **MFAIL** = 1, **JFAIL** must be specified.

89 **ISUBAS**

Subassembly number, only required for the detailed coolant sub-channel model.

90 **JCLN**

Axial heat-transfer segment, **JCLN**, output on the plotting unit for fuel information, 1 ≤ **JCLN** ≤ MZ.

91 **JNEN**

Axial heat-transfer segment, **JNEN**, output on the plotting unit for the cladding temperature, 1 ≤ **JNEN** ≤ MZ.

92 **JNCN**

Axial coolant node, **JNCN**, output on the plotting unit for coolant temperature, 1 ≤ **JNCN** ≤ MZC.

93 **JNSN**

Axial coolant node, **JNSN**, output on the plotting unit for structure temperature, 1 ≤ **JNSN** ≤ MZC.

94 **JRPRO**

Axial heat-transfer segment, **JRPRO**, 1 ≤ **JRPRO** ≤ MZ, output on the plotting unit for total radial profile (written each time the mass averaged fuel temperature, TBAR, increases by **DTFUEL** degrees). Not currently operational.

95 IPSIG

Hydrostatic pressure for fuel swelling:

= 1, Use SIGR.
= 2, Use (SIGR + SIGC)/ 2.
= 3, Use (SIGR + SIGC + SIGZ)/ 3.

96 IHTPRS

= 1, Hot pressing of fuel to **PRSMIN**, not yet operational.
= 0, No hot pressing.

(Recommended **IHTPRS** = 0)

97 IPRD

Controls the amount of DEFORM output in the transient calculation.

(For steady-state control see **NSKIP**)

= 0, Only short form output.
= 1, Radial stress, total porosity, crack volume + short form.
= 2, All above + circumferential stresses retained fission-gas distribution, fission-gas porosity.
= 3, All above + axial stresses, grain size distribution, radial mesh locations.

98-107 IDBFLG

IDBFLG(4) > 0 gives reentry temperature debug print. (10) **IDBFLG**(5) > 2 gives TSCA debug print.

IDBFLG(6) > 0 gives subassembly-subassembly heat transfer prints.

IDBFLG(7) > 0 for film motion of debugs.

IDBFLG(8) > 0 for Wallis flooding correlation debugs.

IDBFLG(9) > 0 for sub-channel analysis coolant debugs

IDBFLG(10) > 0 for sub-channel analysis heat transfer debugs

108-117 IDBSTP

TSCA debug print starts at coolant step **IDBSTP**(5).

118 IEQMAS

Radial fuel mesh size assumption.

= 0, Equal radial distances between points at which temperatures are calculated.
> 0, Equal cylindrical areas associated with each radial temperature node.

119 IBLPRN

Number of coolant dynamics time steps between boiling printouts.

(See **IPO**, **IPOBOI**, and **IBLPRT**).

Default: 100.

120 IDBGBL

IDBGBL > 3 gives boiling debug print.

121 IDBLST

Boiling debug print starts at coolant step **IDBLST**.

122 ISSFU2

= 0, Omit DEFORM calculation during the transient. (See **IAXEXP**).
= 1, Use DEFORM calculation during the transient.
=-1, Simple bond gap conduction model.

123 IHEALC

= 0, No crack healing.
= 1, Crack healing based on fuel swelling rate parameters. (Not operational)
= 2, Cracking healing based on 100% healed if temperature above **FTMPCH***TMF(IFUEL).

124 IAXTHF

Determine components active in the axial expansion calculation in DEFORM.

= 0, Only thermal effects.
= 1, Thermal and force effects.

125 IDCLGO

Value of ICOUNT (number of cladding time steps) when cladding debug print begins.

126 IDCLSP

Value of ICOUNT when CLAP debug print ends.

127 **IDCLDE**

CLAP debug is printed after every **IDCLDE** call to the CLAP module.

128 **IFILM**

Number of nodes dried out before switch from wet (a few boiling segments) to dry (larger boiling length) minimum film thickness.

Suggested value: 3 or 4. Do not input 0.

129-131 **NZONF**

Not currently used.

132-155 **IFUELI**

Not currently used.

156-179 **NODSUM**

Not currently used.

180 **IFUOPT**

Not currently used.

181 **IAXEXP**

Simple axial expansion reactivity feedback model. Can not be used with DEFORM-4. See **ISSFUE** and **ISSFU2**. See **MODEEX** for model choice.

= 0, No simple axial expansion feedback.
= 1, Calculate feedback.
= 2, Calculate and print simple axial expansion feedback for each channel.

182 **IMOMEN**

LEVITATE option referring to the convective momentum flux formulation.

= 0, Central formulation (recommended).
= 1, Upstream formulation.

183 JSTRDX

Axial node number in structure corresponding to the above core load pad. Use only if **IRADEX** = 1,2,3 or -1,-2,-3.

184 IFAE

Fuel adjacency effect in Kramer-DiMelfi cladding failure model.

= 0, No
= 1, Yes

185 ICLADK

Cladding thermal conductivity option.

= 0, Table lookup.
= 1, Functional form cladding conductivity option.

(Not currently operational, use = 0)

186 IFRFAC

= 0, Turbulent and laminar friction factors only.
= 1, Add turbulent and laminar friction factors.
= 2, Turbulent, transition, and laminar friction factors.

AFR, **BFR**, **AFLAM**, **RELAM**, and **RETRAN** are required when **IFRFAC** = 2. Boiling, multiple pin, and subchannel models cannot be used with **IFRFAC** = 2.

187 IRDEXP

Option to use or skip this channel in the radial expansion feedback calculation. Only used when **IRADEX** $\neq$ 0.

= 0, Use this channel.
= 1, Skip this channel.

188 IBUGPN

Debug flag, should currently always be 0.

189 IMETAL

Indicates fuel type.

= 0, Oxide fuel.
> 0, Metal fuel.
= 1, Uranium-fissium metal fuel.
= 2, U-Pu-Zr ternary alloy fuel.
= 3, U-Zr binary alloy fuel.

Note that MFUEL (**IFUELO**= 2) requires **IMETAL**= 2 or 3.

190 IPNPLT

=1, The PINACLE module produces a printer plot of the axial fuel distribution with every full printout.

191 IFUELO

Option for annular zone formation model in U-Pu-Zr alloy fuel (for the case of **IMETAL** = 2 only).

= 0, User input zonal compositions and radii (See **IFUELC**, **IZNC**, **IZNM**, **MFTZN**, **RIZNC**, and **RIZNM**. See also **PUZRTP**, **RHOZN**, **XLOGNA**, **PRSTY**, **TFSOL**, and **TFLIQ**).
= 1, (Not functional) Zonal composition and radii computed using the SSCOMP physical model (See **TTRANM**, **TTRANC**, **POROSS**, **PORMSS**, **PORCSS**, **FPORNA**, **RHOREF**, **WUREF**, **WPUREF**, **WZRREF**, **PUBYU**, **CPCM**, **CPMO**, **CZCM**, **CZMO**, **CUCM**, **CUMO**, **EPSMS**, **EPSCOM**, **IDSSC**, and **BURNFU**).
= 2, Fuel composition and geometry computed by MFUEL physical model.

192 IFUELM

Option for the thermal properties of the U-Pu-Zr alloy fuel (for the case of **IMETAL** > 1 only).

= 0, Use properties interpolated from the IFR metallic fuels handbook data.
= 1, Use properties in the report ANL/RAS 85-19.
= 2, Use Mark-V fuel or U-10Zr fuel properties (depending on **IMETAL** = 2 or 3) based on the metallic fuels handbook data.
= 3, Use thermal conductivity correlation recommended for use with MFUEL (MFUEL must be active, **IFUELO**= 2).

193 IFUELC

User input fuel zone specification flag.

= 0, Single radial fuel zone (See **IFUELV**, **IFUELB**, **MZLB**, **MZUB**).
= 1, Multiple radial fuel zones (See **IZNC**, **IZNM**, **MFTZN**, **RIZNC**, **RIZNM**).
= 2, Each radial mesh interval is a unique fuel zone for **IMETAL** > 1, with input mesh point porosities and compositions determined in interface routine LIFEIF.

194 **IPNGO**

Between cycles **IPNGO** and **IPNSTP** a full PINACLE output is obtained at cycle intervals containing **IPNNEW** cycles.

195 **IPNSTP**

See **IPNGO**.

196 **IPNNEW**

See **IPNGO**.

197 **IEUTOPT**

MFUEL eutectic penetration option.

= 0, Mechanistic eutectic penetration, default axial expansion model
= 1, Empirical eutectic penetration, default axial expansion model
= 2, Mechanistic eutectic penetration, conservative axial expansion model
= 3, Empirical eutectic penetration, conservative axial expansion model

198-202 **IDM51**

Not currently used.

203 **IDKCRV**

Power or decay heat curve for this channel.

204 **ITP20**

Fuel, cladding, and coolant temperatures and voiding data are written on unit 20 every **ITP20** time steps. If **ITP20** ≤ 0, no unit 20 output.

CHANNEL-TO-CHANNEL HEAT TRANSFER

205 **NCHCH**

Number of other channels that this channel is in contact with for duct wall-to-duct wall heat transfer. Maximum 8.

If **NCHCH** < 0, Q(ICH to JCH) = -Q(JCH to ICH)

206-213 **ICHCH**

Channel number of the K-th channel that this channel is in contact with. See also **HACHCH**.

If **ICHCH** < 0, then -**ICHCH** is a bypass channel number.

If **ICHCH**(1) < -8, then -**ICHCH**(K) is the temperature of a constant temperature heat sink in axial zone K.

If **ICHCH** > 500000, transfer heat from structure of ICH to coolant of **ICHCH** - 500000.

If **ICHCH** > 750000, transfer heat from coolant of ICH to coolant of **ICHCH** - 750000.

MULTIPLE PIN OPTION

214 **JJMLTP**

Multiple pin option.

=0, No multiple pin treatment for this subassembly. Use single-pin model.
>0, This is the first of **JJMLTP** channels used to represent the subassembly.
<0, This is one of the additional channels used to represent the subassembly.

Notes:

1) **JJMLTP** and **ICHCH** refer to different phenomena: Intra-subassembly heat transfer and inter-subassembly heat transfer.

2) **JJMLTP** is used for intra-subassembly coolant-to-coolant heat transfer from channel I to I + 1 and from I to I - 1. **UACH1** and **UACH2** must be supplied to determine the heat transfer coefficients.

3) If M channels are used to represent a subassembly, then they must be consecutive channels, starting with channel ICHN and going to channel ICHN + M - 1. For channel ICHN, **JJMLTP** = M. For channels ICHN + 1 through ICHN + M - 1, **JJMLTP** is negative.

4) The maximum value of M is 56 (no more than 56 channels can be used to represent a subassembly).

5) The axial zones outside the pin section go only with the first channel (channel ICHN). In these zones, the coolant flow area per pin and the reflector and structure perimeters per pin must be based on the number of pins in channel ICHN. The reflector zones are ignored for channels ICHN + 1 to ICHN + M - 1.

6) **ICHCH** refers to subassembly-to-subassembly heat transfer from the current channel to channel **ICHCH**. This heat transfer is from outer surface node to outer surface node. **ICHCH** > 1000 is an

exception. This exception was included for modeling the thimble flow region of the XX09 subassembly in EBR-II. If **ICHCH** ≠ 0 then the heat transfer coefficient times area per unit height is specified by **HACHCH**.

DETAILED COOLANT SUB-CHANNEL MODEL

215-218 **JCHMPN (K)**

Other channel this coolant sub-channel is in contact with. Used only if **ISCH** = 1, **JJMLTP** ≠ 0

Reserved for the detailed coolant sub-channel model.

219 **NUMKLT**

Number of other channels this coolant sub-channel is in contact with. Used only if **ISCH** = 1, **JJMLTP** ≠ 0

Reserved for the detailed coolant sub-channel model.

220 **KSWIRL**

Value of K in **JCHMPN**(K) for swirl flow. Normally **KSWIRL**=0 except in an edge or corner sub-channel. Used only if **ISCH** = 1, **JJMLTP** ≠ 0

Reserved for the detailed coolant sub-channel model.

221 **NULST1**

Number of time steps in a null transient for this subassembly or group of channels. Used only if **ISCH** = 1, **JJMLTP** ≠ 0.

Reserved for the detailed coolant sub-channel model.

SSCOMP METAL FUEL BEHAVIOR MODEL

222 **IPORC**

Options to control open porosity considerations in metal fuel axial swelling during transient calculation.

= 0, Close open porosity by fraction given in **FCLOP** before axial movement takes place.
= 1, Treat open porosity like closed porosity.

223 **IDSSC**

MFUEL (**IFUELO**= 2) detailed print option.

= 0, print only at end of steady-state.
= 1, print at every coarse time step specified by the user.

END SSCOMP METAL FUEL BEHAVIOR MODEL

224 **IDRY**

= 1, For film motion.

225 **ICTYPE**

Cladding material indicator.

= 1, 316
= 2, D9
= 3, HT9
= 4, 15-15Ti

`ICTYPE` = 1, 2, 3, and 4 can be used for MFUEL. `ICTYPE` = 1, 2, and 3 can be used for DEFORM-5 and FPIN. `ICTYPE` = 1, and 4 can be used for DEFORM-4, the default cladding is 316.

SUBASSEMBLY-TO-SUBASSEMBLY HEAT TRANSFER

226-233 **IOPCHC (K)**

Subassembly-to-subassembly heat transfer option.

= 0, Q(JCH,ICH) = - Q(ICH,JCH)
= 1, Q(JCH,ICH) not set by Q(ICH,JCH)

234 **iCSwellID**

Function ID for user-defined void swelling rate. Can be used by MFUEL and DEFORM-4.

235 **iCIrrCreepID**

Function ID for user-defined irradation creep rate. Can be used by MFUEL and DEFORM-4.

236-249 **KTRANC**

Not currently used.

250-273 **KTRANM**

Not currently used.

274 **IPRSKP**

Skip the prints for channel ICH if `IPRSKP` is nonzero. To be used to reduce the volume of printed output from TSPRNT.

275 **ITREAT**

For use with oxide fuel TREAT experiment analysis.

= 1, Completely crack fuel at start of transient.

276 **IOPPL**

Not currently used.

277 **IBUBND**

= 0, Form bubbles at node mid-points.
= 1, Form bubbles at node boundaries.

278 **IGASRL**

= 0, No gas release.
= 1, Gas release in the boiling model, requires DEFORM-5 life fraction calculation and **FRUPT**(1).
= 2, Gas release in the boiling model by pin group at the times given by **TMFAIL** at axial node **INDFAL**.

279 **IGRLTM**

Not currently used.

280 **IRAPEN**

Rapid eutectic formation rate assumption.

= 0, Use exponential eutectic formation rate formula for all temperatures (i.e. no eutectic formation temperature threshold and no rapid eutectic penetration allowed).
= 1, Allow rapid eutectic penetration if molten fuel is available (i.e. cladding temperature above fuel solidus temperature) and temperatures are in rapid penetration region (1353K to 1506K).
> 1, Use exponential eutectic formation rate formula above 923K (650 deg. C) and allow rapid eutectic penetration in the range from 1353K to 1506K (EBR-II Mark-V safety case).

281 **LCHTYP**

Core channel designator.

= 1, Fuel channel.
= 2, Reflector channel.
= 3, Control rod channel.

282 **IGSPRS**

Controls fission gas assumptions used with metal fuel pins.

= 0, Equilibrium maintained between plenum and axial segments of fuel by instantaneous gas relocation.
= 1, After initiation of equilibrated pressure during first transient time stop, gas remains in axial segments and may locally pressurize.

283 **INDFAL**

Failure node number on the fuel pin axial mesh for pin failure and gas release if **IGASRL**=2.

284 **IAXCON**

Axial coolant heat conduction option.

= 0, No axial heat conduction in the coolant.
= 1, Axial heat conduction in the coolant. Only operative with the multiple pin option (**JJMLTP** not equal to 0).

FPIN-2 INPUT

285 **IFPIN2**

= 0, Do not use FPIN2 metal fuel model.
= 1, Use FPIN2 metal fuel model.

No other data required when **IFPIN2**=0.

286 **IFPI01**

= 0, Use FPIN2 in interfaced mode.
= 1, Use FPIN2 in stand-alone mode.

No other data required when **IFPI01**=1.

287 **IHTFLG**

= 0, Bypass FPIN2 heat transfer calculation.
= 1, Include FPIN2 heat transfer calculation (for debugging purposes only).

288 **LHTOPT**

= 0, Perform heat transfer calculation including coolant and structure.

= 1, Perform heat transfer calculation with input values of cladding outer surface temperature.

This input is required only for **IHTFLG**=1.

289 LCRACK

Fuel cracking option.

= 0, No cracking.
= 1, Radial cracks included.

290 LFPLAS

Option for creep-plastic strains in fuel.

= 0, Allow creep-plastic strains.
= 1, Suppress creep-plastic strains.

291 LCPLAS

Option for creep-plastic strains in cladding.

= 0, Allow creep-plastic strains.
= 1, Suppress creep-plastic strains.

292 LFSWEL

Option for swelling-hotpressing strains in fuel.

= 0, Allow swelling-hotpressing strains.
= 1, Suppress swelling-hotpressing strains.

293 LCSWEL

Option for swelling strains in cladding.

= 0, Allow swelling strains.
= 1, Suppress swelling strains.

294 **LLRGST**

Option for strain analysis.

= 0, Large strain analysis.
= 1, Small perturbation analysis.

295 **LFCSLP**

= 0, Fuel-cladding locked when gap is closed.
= 1, Independent fuel-cladding axial displacement.

296 **LOUTSW**

Print option.

= 0, No detailed printing of results - summary only.
= 1, Normal detailed printout under **LFREQA**, **MFREQA**, and **LFREQB** control.

297 **LFREQA**

Initial print frequency, number of time steps between normal detailed printout.

298 **MFREQA**

Total number of time steps under **LFREQA** control.

299 **LFREQB**

Final print frequency.

300 **LGRAPH**

Graphics file option.

= 0, Do not write graphics file.
= 1, Write a graphics datafile.

301 **LDBOUT**

= 0, No debug output.
= 1, Add debug output to regular **LOUTSW**=2 output.

302 LDBSTP

= 0, FPIN2 calculation stops when molten cavity freezes.
= 1, Ignore this program stop.

303 LDBFPL

= 0, Use recommended fuel flow stress.
= 1, Simple power law fuel creep (EPSDOT=C0*SIGE**C1).

See **XFPLC0** and **XFPLC1**.

304 LDBFDV

= 0, Use recommended fuel swelling - hot pressing. (Fuel swelling option for metal fuel is the simple grain boundary swelling model, ANL-IFR-27 and ANL/RAS 83-33).
= 1, Use equilibrium swelling model (ANL-IFR-6 & -23).
= 2, Simple power law fuel swelling (EPVDOT=C0*SIGM**C1).

See **XFDVC0** and **XFDVC1** .

305 LDBCPL

= 0, Use recommended cladding flow stress.
= 1, Ideal plastic flow for cladding (SIGY=C0+C1*EPBAR).

See **XCIPL0** and **XCIPL1** .

= 2, Use high-temperature power-law creep.
= 3, Use simple power law cladding creep

(EPSDOT=C0*SIGE**C1).

See **XCIPL0** and **XCIPL1** .

306 LGPRES

Not currently used.

307 LGAPCL

= 0, Use fuel-cladding opening/closure model.
= 1, Fuel-cladding gap always closed.

308 LCPROP

= 0, Use material property correlations.
= 1, Use temperature independent material properties.

(SAS thermal properties are used for **IFPI01**=0).

309 **LSKIPM**

= 0, Perform mechanical calculations.
= 1, Bypass mechanical calculations.

310 **LGCLOS**

= 0, Use gap closure routine at 100% fuel melting.
= 1, Do not close gap (if open) at 100% fuel melting.

311-334 **LDBOTA (J)**

Axial debug print vector.

= 0, No print.
= 1, Print.

335-345 **LDBOTF (IF)**

Fuel radial debug print vector.

= 0, No print.
= 1, Print.

346-348 **LDBOTC (IC)**

Clad radial debug print vector.

= 0, No print.
= 1, Print.

349-360 **NONE**

Reserved.

PRIMAR-4 MULTIPLE INLET/OUTLET PLENA

361 **NSEGMP**

PRIMAR-4 segment number to which this channel is assigned in the multiple inlet/outlet plena model (See **IFMIOP**).

CONTROL ROD DRIVE FEEDBACK

362 **ICHUIS**

= 0, Coolant from channel ICH is included in the upper internal structure temperature calculation for control rod drive expansion reactivity.
= 1, This channel is not used.

PINACLE

363 **LQSLTP**

= 0, Wall friction is not considered in calculating the velocity of the sodium slug above the fuel.
= 1, Wall friction is considered.

POWER AND REACTIVITY MESH

364 **IPOWRZ**

Axial power shape input option.

= 0, Enter channel axial power shape in array **PSHAPE** on the MZ axial mesh; JMAX=24.
= 1, Enter channel axial power shapes for core and axial blanket fuel in arrays **PSHAPC** (core fuel power shape) and **PSHAPB** (blanket fuel power shape) on the MZC axial mesh; JMAX=48.

365 **IREACZ**

Axial reactivity worth input option.

= 0, Enter channel axial reactivity worth for Doppler, coolant void, cladding motion, fuel motion, and structure motion in arrays **WDOPA**, **VOIDRA**, **CLADRA**, **FUELRA**, and **STRCRA** on the MZ axial mesh; JMAX=24.
= 1, Enter channel axial reactivity worth for Doppler, coolant void, cladding motion, fuel motion, and structure motion in arrays **WDOPA**, **VOIDRA**, **CLADRA**, **FUELRA**, and **STRCRA** on the MZC axial mesh; JMAX=48.

FUEL ZONE TYPE ASSIGNMENT

366-389 **IZNC (J)**

Outermost radial mesh interval of central zone at axial segment J. Maximum value = **NT**. See also **RIZNC**. Used only for **IFUELC**= 1.

390-413 **IZNM (J)**

Outermost radial mesh interval of middle (intermediate) zone at axial segment J. Maximum value = **NT**. See also **RIZNM**. Used only for **IFUELC**= 1.

414-485 **MFTZN (L,J)**

Fuel type (IFUEL) assignment to radial zones at axial segment J. Maximum value = 8. Used only for **IFUELC**= 1.

L=1, fuel type assigned to central (inner) zone. L=2, fuel type assigned to middle intermediate) zone. L=3, fuel type assigned to outer zone.

Note: A maximum of three radial zones may be specified at each axial level. Fewer than three zones may also be specified, with a minimum of a single zone assigned to all the fuel in the pin at a given axial location. Zones are assigned assuming azimuthal symmetry; the central (inner) zone begins at the fuel centerline and extends outward radially through radial mesh interval **IZNC**(J). The middle (intermediate) zone begins at radial mesh interval **IZNC**(J)+1 and extends outward radially through radial mesh interval **IZNM**(J). The outer zone begins in radial mesh interval **IZNM**(J)+1 and extends outward radially to the fuel surface (radial temperature node **NT**). The central zone may be eliminated by setting **IZNC**(J) = 0. The middle zone may be eliminated by setting **IZNM**(J) = **IZNC**(J). The outer zone may be eliminated by setting **IZNM**(J) = **NT**. Both inner and middle zones may be eliminated by setting **IZNC**(J) = **IZNM**(J) = 0. The central zone is present only if **IZNC**(J) > 0. The middle zone is present only if **IZNM**(J) > **IZNC**(J). The outer zone is present only if **IZNM**(J) < **NT**. The **MFTZN** array assigns a fuel type to a zone; fuel types and zones have a one- to-one correspondence at each axial level. The fuel types assigned here specify the fuel thermo-physical properties to be used in the solution of the fuel pin heat transfer equations. See **IFUELC**.

486 **IPINFG**

Metal fuel fission gas model flag.

= 0, Use DEFORM-5 fission gas formulation.
> 0, Use PINACLE fission gas formulation.

487 **IPINRE**

PINACLE ejected fuel re-entry flag.

= 0, Re-entry of fuel ejected into above-core region not permitted.
> 0, Allow re-entry into active core region of fuel ejected into above-core region.

488 **IPORFG**

Density correction to porosity in DEFORM-5 fission gas model.

= 0, Neglect transient temperature/density impact on open porosity volume.
> 0, Adjust open porosity volume accounting for transient temperature/density changes.

489 **IPRSS1**

Steady state fuels characterization initial values print in subroutine SSINC1.

= 0, Print initial values for pointwise fuel mass and porosity; Pu content for **IMETAL** = 2; Zr content for **IMETAL** = 2 or 3; Fe and Ni content for **IMETAL** = 2 and **IFUELO** = 1; Na content for **IMETAL** > 0.
= 1, No prints.

490 **IMKVPL**

EBR-II Mark-V safety case plotting data.

= 0, Do not save data for post-processing.
> 0, Compute and save maximum fuel temperature in low Zr zone, maximum inner cladding temperature, and minimum coolant saturation temperature in the channel on each main time step (Entry TSPLT1 in subroutine TSPLOT). Compute and save reactor power, flow, and power-to-flow ratio on each main time step (Entry TSPLT3 in subroutine TSPLOT).

491 **MZCHCH**

See FED.

492 **KZEMFM**

See FED.

493 **MTREAT**

TREAT fuel channel modeling flag.

= 0, Use standard fuel channel models.
> 0, Use special TREAT fuel channel models. See also **CFLAT** and **FFLAT**. **RBR**, **RER**, and **ROUTFP** contain TREAT fuel assembly half-thickness dimensions. In fuel and cladding heat transfer calculations, the correct periphery will be used in place of the circumference. The structure field will be eliminated. The fission gas plenum will be eliminated. Air properties will be used for the coolant. The coolant pressure drop calculation will be eliminated. One pin per channel will be used. The DEFORM, PRIMAR, and multiple pin models are not allowed. Only equally-spaced radial heat transfer mesh (**IEQMAS**=0) is allowed.

494 **IFLOOD**

= 0, Use **FVAPM** to determine the friction factor multiplier for the vapor.

= 1, Use Wallis flooding correlation to determine whether to use the Wallis friction factor multiplier for the vapor. This option should only be used with the film motion model, **IDRY** = 1.

DETAILED COOLANT SUB-CHANNEL MODEL

ISCH = 1 and **JJMLTP** ≠ 0

495 **IFT24**

Output detailed coolant sub-channel model variables on fort.24 every **IFT24** time steps. If **IFT24** = 0, no output on fort.24.

Reserved for the detailed coolant sub-channel model

496 **ILATF**

Always use **ILATF** = 0 to include lateral flow terms in the momentum equation for the detailed coolant sub-channel model.

Reserved for the detailed coolant sub-channel model.

497 **IDEFOPT**

= 0, use default DEFORM-4 models

> 0, use extended DEFORM-4 models.

Note that the extended DEFORM-4 models will be fully available at later versions. Currently, only 15-15Ti cladding models (**ICTYPE** = 4) are supported.

498 **IDEFSTFAL**

= 0, do not use stochastic clad damage evaluation model

> 0, use stochastic clad damage evaluation model as a sidebar analysis. Only applicable when **IDEFOPT** > 0 and **ICTYPE** = 4.

499 **IDMICH**

Not currently used.

500 **LDETL**

Visualization Data Detail Flag:

= 0, Do not write visualization data for this channel
= 1, Write coolant temperatures only
= 2, Coolant and duct wall (structure) temperatures
= 3, Coolant, structure, and bulk fuel temperatures
= 4, Coolant, structure, average fuel, and detailed radial fuel temperatures.

501 **NFT24**

Number of items to be output on fort.24

Reserved for the detailed coolant sub-channel model.

502-521 **JCFT24 (K)**

Axial node for the Kth output on fort.24.

Reserved for the detailed coolant sub-channel model.

522-541 **ITYP24 (K)**

Variable type for the Kth output on fort.24

= 1, PCBAR2 (coolant pressure, middle of the node)
= 2, W2RT2 (coolant flow rate)
= 3, TCOOL2 (coolant temperature at the bottom of the node)
= 4, TCBAR2 (coolant temperature at the middle of the node)
= 5, PCOOL2 (coolant pressure, bottom of the node)
= 6, WLAT2(1) (coolant lateral flow rate to first adjacent subchannel
= 7, WLAT2(2) (coolant lateral flow rate to second adjacent subchannel
= 8, WLAT2(3) (coolant lateral flow rate to third adjacent subchannel
= 9, WLAT2(4) (coolant lateral flow rate to fourth adjacent subchannel
=10, NITER (number of iterations for the last solution in SOLVIT

Reserved for the detailed coolant sub-channel model.

542 **NULPT1**

Print results every **NULPT1** steps for the null transient (see **NULST1**)

Reserved for the detailed coolant sub-channel model.

543 **iReactivityTable**

= 0, Use existing **FUELRA**, **CLADRA**, **VOIDRA**, and **STRCRA** input locations.

> 0, Channel-dependent *Table ID* referencing table-based reactivity feedback data. Optional columns named *FUELRA*, *CLADRA*, *VOIDRA*, or *STRCRA* in this table will supersede any corresponding input in the **FUELRA**, **CLADRA**, **VOIDRA**, or **STRCRA** input locations.

If the channel-dependent table does not exist, a global table must exist with the corresponding ID. *Example input* is shown along with the input definition in Table 4.5.1.

Currently, **iReactivityTable** > 0 cannot be used with the sodium boiling, in-pin or ex-pin fuel motion, or cladding motion models.

544 **iMassFractionTable**

= 0, Fuel consists of a single component whose reactivity is defined by **FUELRA** or by column *FUELRA* in **iReactivityTable**.

> 0, Channel-dependent *Table ID* referencing table-based mass fraction component descriptions for fuel. Component reactivity is defined in Table **iReactivityTable**. If only a single component is defined and the component label is *FUELRA*, the component reactivity may be defined by **FUELRA**.

If the channel-dependent table does not exist, a global table must exist with the corresponding ID. *Example input* is shown along with the input definition in Table 4.5.2.

Currently, **iMassFractionTable** > 0 cannot be used with the sodium boiling, in-pin or ex-pin fuel motion, or cladding motion models.

545 **ZoneLossCoefTableID**

= 0, Anisotropic loss coefficients are not used.

> 0, Channel-dependent *Table ID* referencing anisotropic Re-dependent loss coefficient data. This table can be defined globally or within the *CHANNEL* block.

If the channel-dependent table does not exist, a global table must exist with the corresponding ID. *Example input* is shown in Section 3.9.3.1.

546-600 **IDMICH**

Not currently used.

2.7.12 Block 61 — GEOMIN — Channel-Dependent Geometry Data

1-7 **ACCZ (KZ)** **m^2**

Coolant flow area per fuel-pin in zone KZ. For example, core plus axial blankets plus fission-gas plenum, i.e., the whole fuel-pin length, is one zone. Each axial reflector is a separate zone.

8-31 **AXHI (J)** **m**

Length of axial segment J, 1 ≤ J ≤ MZ. MZ is the number of axial segments in core plus blankets. Note: All should be as equal in size as possible. MFUEL (**IFUELO**= 2) requires this dimension to be defined based on as-fabricated dimensions.

32-38 **DHZ (KZ)** **m**

Hydraulic diameter in zone KZ.

39-45 **DSTIZ (KZ)** **m**

DSTI (thickness of inner structure node) in zone KZ. (See **FNSROS** for consistent input).

46-52 **DSTOZ (KZ)** **m**

DSTO (thickness of outer structure node) in zone KZ. (See **FNSROS** for consistent input)

53 **PLENL** **m**

Length of fission-gas plenum. Note: MFUEL (**IFUELO**= 2) requires this dimension to be defined based on as-fabricated dimensions.

54-77 **RBR (J)** **m**

Cladding inner radius for axial segment J, 1 ≤ J ≤ MZ.

Note: If all values are the same, only the first must be input. MFUEL (**IFUELO**= 2) requires this dimension to be defined based on as-fabricated dimensions.

78-101 **RER (J)** **m**

Cladding outer radius for axial segment J, 1 ≤ J ≤ MZ.

Note: If all values are the same, only the first must be input. MFUEL (**IFUELO**= 2) requires this dimension to be defined based on as-fabricated dimensions.

102 **RBRPL** **m**

Cladding inner radius in gas plenum. Note: MFUEL (**IFUELO**= 2) requires this dimension to be defined based on as-fabricated dimensions.

103 **RERPL** **m**

Cladding outer radius in gas plenum. Note: MFUEL (**IFUELO**= 2) requires this dimension to be defined based on as-fabricated dimensions.

104-127 **RINFP (J)** **m**

Fuel inner radius for axial segment J, 1 ≤ J ≤ MZ. (Minimum of 1.0E-6 if **ISSFUE** = 1).

Note: If all values are the same, only the first must be input. MFUEL (**IFUELO**= 2) requires this dimension to be defined based on as-fabricated dimensions.

128-151 **ROUTFP (J)** **m**

Fuel outer radius for axial segment J, 1 ≤ J ≤ MZ. Note: If all values are the same, only the first must be input. MFUEL (**IFUELO**= 2) requires this dimension to be defined based on as-fabricated dimensions.

152-158 **ZONEL (KZ)** **m**

Length of zone KZ.

159-165 **SRFSTZ (KZ)** **m**

SFR, structure perimeter in zone KZ per pin.

166 **AREAPC** m^2

APC, coolant plus pin area per pin, in the pin section.

167 **CFLAT** m

Beveled corner dimension on outer cladding surface for **MTREAT** > 0.

168 **FFLAT** m

Beveled cormer dimension on outer fuel surface for **MTREAT** > 0.

169-175 **DRFO (KZ)** m

KZ = 1, NZONE. For KZ not equal to KZPIN , **DRFO** is the thickness of the outer reflector node in zone KZ. Note that a two-node, slab-geometry treatment is used for heat transfer to the reflector. (See **DRFI** for the inner node thickness). For KZ = KZPIN, **DRFO** = cladding thickness in the plenum region (**RERPL** - **RBRPL**).

176 **STCOR**

Not currently used.

177 **VFPLNT**

Not currently used.

178 **VFLREF**

Not currently used.

179 **VFUREF**

Not currently used.

180 **RBR0** m

Nominal cladding inner radius.

181 **RER0** m

Nominal cladding outer radius.

182-188 **SER (KZ)** m

For KZ not equal to KZPIN, **SER** is the reflector perimeter per pin wetted by coolant in the reflector zone. For KZ = KZPIN, **SER** is the pin perimeter in the gas plenum region.

189-195 **DRFI (KZ)** **m**

KZ = 1,…,NZONE. **DRFI** is the thickness of the inner reflector node in zone KZ.

196 **VFC**

Not currently used.

197-220 **RAFUZ**

Not currently used.

221 **ZCHOBT**

Axial coordinates lower end of the hodoscope field of view referenced to the fuel pin bottom. Used only for TREAT experiment analysis.

222 **ZCHOTP**

Axial coordinates upper end of the hodoscope field of view referenced to the fuel pin bottom. Used only for TREAT experiment analysis.

223 **ZOFFST** **m**

Axial (Z) offset, elevation of the bottom of the lower blanket. Default = 0.

224-247 **RIZNC (J)** **m**

Outer radius of central zone corresponding to **IZNC**(J). Default value of zero results in central zone outer radius being set by **IZNC**(J) and the assumption of an equal delta R radial mesh. Used only for **IFUELC** = 1.

248-271 **RIZNM (J)** **m**

Outer radius of intermediate zone corresponding to **IZNM**(J). Default value of zero results in intermediate zone outer radius being set by **IZNM**(J) and the assumption of an equal delta R radial mesh. Used only for **IFUELC** = 1

272 **TWASTI** **m**

Initial wastage thickness on the cladding inner surface. Used in DEFORM-5 and FPIN2.

273 **TWASTO** **m**

Initial wastage thickness on the cladding outer surface. Used in DEFORM-5 and FPIN2.

274 **XXNPIN**

Number of pins per subassembly. Used only by the multiple pin model. See also **NPIN**. If **XXNPIN** is 0.0, then the code will set **XXNPIN** = **NPIN** . If **XXNPIN** is not 0.0, then **NPIN** will be set to **XXNPIN**, rounded (not truncated) to the nearest integer.

275-284 **DZEMFM**

See FED.

285-350 **DUMGEO**

Not currently used.

2.7.13 Block 62 — POWINC — Channel-Dependent Power and Reactivity Input

1 **FPDAYS**

Effective full power days during the steady state irradiation. Used with **FLTPOW** to determine cladding fluence when **BURNFU** is not used.

2 **GAMSS**

Fraction of total power in direct heating of structure.

3 **DUM002**

Not currently used.

4 **GAMTNC**

Fraction of total power in direct heating of coolant.

5 **GAMTNE**

Fraction of total power in direct heating of cladding.

6-29 **PSHAPE (J)**

Ratio of pin power at axial segment J to the power **POW** in the peak axial fuel pin segment, $1 \leq J \leq MZ$. MZ is the number of axial heat transfer segments in the core plus axial blankets. Enter only for **IPOWRZ** = 0. The first entered segment value (J=1) is for the lower-most segment in the lower axial blanket. Values are normalized to **POW** or **POWTOT** using **NPIN**, **NSUBAS**, and **PRSHAP** at the initial steady-state.

30-44 **PSHAPR (I)**

Radial power shape within pin by radial node, I, normalized power per unit fuel mass, $1 \leq I \leq$ **NT**.

45-52 **PLIN (M)**

When **MSTEP**≤ 8, relative power level at the end of the M steady-state division. It is required that **PLIN** is 1.0 at the end of the steady-state. It is assumed that during power change periods, the power

varies linearly over the time interval from the previous value to the current **PLIN**(M) value. (M = 1, **MSTEP**)

53-60 **TPLIN (M)**

When **MSTEP** ≤ 8, time (from fresh start-up) at the ends of the M steady-state divisions (seconds). M = 1, **MSTEP**.

61 **FLTPOW** **(\#/m^2-s)/(W/m)**

Fast flux to linear power ratio.

62 **ADOP** **Δk/k**

Doppler coefficient for this channel when part of the core represented by this channel is not voided.

63 **BDOP** **Δk/k**

Doppler coefficient for this channel when part of the core represented by this channel is fully voided.

64-111 **WDOPA (J)**

Doppler axial weighting factor. Enter MZ values for **IREACZ**=0. Enter MZC-1 values for **IREACZ**=1.

112-159 **VOIDRA (J)** **Δk/k-kg**

Coolant void reactivity worth per unit coolant mass. Enter MZ values for **IREACZ**=0. Enter MCZ-1 values for **IREACZ**=1.

160-207 **CLADRA (J)** **Δk/k-kg**

Cladding reactivity worth per unit cladding mass. Enter MZ values for **IREACZ**=0. Enter MCZ-1 values for **IREACZ**=1.

208-255 **FUELRA (J)** **Δk/k-kg**

Fuel reactivity worth per unit fuel mass. Enter MZ values for **IREACZ**=0. Enter MCZ-1 values for **IREACZ**=1.

256 **PRSHAP**

Ratio of power per subassembly averaged over this channel to the power per sub- assembly averaged over all channels.

Note: These values are normalized by the code over all assemblies such that the average is 1.0.

257-261 **PSHPTP (J)**

In PINACLE, PLUTO2, and LEVITATE for **IPOWRZ**=0, ratio of active fuel power at axial segment KCORE2+J to the power **POW** in the peak axial fuel segment. 1 ≤ J ≤ 5. KCORE2 is the top active

fuel node. This array is used when active fuel relocates above the original active fuel, either in-pin (PINACLE) or ex-pin (PLUTO2, LEVITATE).

262-266 **PSHPBT (J)**

In PLUTO2 and LEVITATE for **IPOWRZ**=0, ratio of active fuel power at axial segment KCORE1 - J to the power **POW** in the peak axial fuel segment. 1 ≤ J ≤ 5. KCORE1 is the bottom active fuel node. This array is used when active fuel relocates below the original active fuel due to ex-pin fuel motion (PLUOT2, LEVITATE).

267 **FLOWBU**

Fraction of subassemblies in channel that have low (<2.9 at.%) burnup.

268 **XRFSHP**

Relative reactivity worth of fuel in the channel. Used in EBR-II feedback model.

269 **XRNSHP**

Relative reactivity worth of sodium in the channel. Used in EBR-II feedback model.

270 **XRSSHP**

Relative reactivity worth of stainless steel in the channel. Used in EBR-II feedback model.

271-318 **PSHAPC (J)**

Ratio of core fuel pin power at axial segment J to the power **POW** in the peak axial fuel pin segment, 1 ≤ J ≤ MZC-1. MZC-1 is the number of axial coolant segments. Enter only for **IPOWRZ** = 1. The first entered segment value (J=1) is for the lower-most segment in the lower axial reflector. Values are normalized to **POW** or **POWTOT** using **NPIN**, **NSUBAS**, **PRSHAP**, **PSHAPB**, and the core fuel and blanket fuel mass distributions at the initial steady-state.

319-366 **PSHAPB (J)**

Ratio of blanket fuel pin power at axial segment J to the power **POW** in the peak axial fuel pin segment, 1 ≤ J ≤ MZC-1. MZC-1 is the number of axial coolant segments. Enter only for **IPOWRZ** = 1. The first entered segment value (J=1) is for the lower-most segment in the lower axial reflector. Values are normalized to **POW** or **POWTOT** using **NPIN**, **NSUBAS**, **PRSHAP**, **PSHAPC**, and the core fuel and blanket fuel mass distributions at the initial steady-state.

EBR-II MK-V SAFETY CASE INPUT

367 **FHTCLD**

Cladding steady-state hot channel factor.

368 **FUNKCL**

Transient cladding thermal conductivity uncertainty factor.

369 **FUNKFU**

Transient fuel thermal conductivity uncertainty factor. Used for **IMETAL** > 0 and **IFUELM** = 2.

370 **FUNFLM**

Transient film heat transfer uncertainty factor.

371 **FHTFUL**

Fuel steady-state hot channel factor.

372 **FUNCOL**

Transient coolant flow uncertainty factor.

373 **FPKNG**

Peaking factor.

374 **FUNPOW**

Transient power uncertainty factor.

END OF EBR-II MK-V SAFETY CASE INPUT

375 **COILP0**

See FED.

376 **COILP1**

See FED.

377 **COILP2**

See FED.

378-425 **STRCRA (J)** **Δk/k-kg**

Structure reactivity worth per unit structure mass. Enter MZ values for **IREACZ**=0. Enter MCZ-1 values for **IREACZ**=1.

2.7.14 Block 63 — PMATCH — Channel-Dependent Properties Input

1 TIMCNS — **s**

Time constant for fuel pin heat transfer. Default = 0.1.

2 AHBPAR — **W/m^2-K\ \ \ m^2-K/W-m**

If all three parameters are not equal to 0.0, the bond correlation is of the form:

HB = **AHBPAR** + 1.0/(**BHBPAR** +(gap size + **CHBPAR**)/**HBPAR**)

3 BHBPAR

see **AHBPAR**

4 CHBPAR

see **AHBPAR**

5 HBMAX — **W/m^2-K**

Maximum value of bond conductance when a gap exists; minimum value when a gap does not exist.

Default when **AHBPAR**, **BHBPAR**, and **CHBPAR** > 0.0:

HBMAX = **AHBPAR**
+ 1.0/(**BHBPAR**+**CHBPAR**/**HBPAR**)

6 HBMIN — **W/m^2-K**

Minimum value of bond conductance.

7 HBPAR — **W/m-K**

If all three parameters in **AHBPAR**, **BHBPAR**, and **CHBPAR** are zero, then the bond conductance equation when a gap exists is of the form: Bond conductance = **HBPAR**/gap.

8 HSNK

Should be set to 0.0.

9 TSNK

Should be set to 0.0.

10 **DPRSTY**

Not currently used.

11-17 **XKSTIZ (KZ)** **W/m-K**

Inner structure thermal conductivity in zone KZ.

18-24 **XKSTOZ (KZ)** **W/m-K**

Outer structure thermal conductivity in zone KZ.

25 **DEL** **W/m^2-K^4**

Stefan-Boltzmann's constant*emissivity. (See **STEBOL**, **EMSF**, and **EMSC**).

26 **DGO** **m**

Initial grain size. Suggested value: 1.0E-05.

27 **P0GAS** **Pa**

Initial plenum gas pressure at reference temperature TR.

28-34 **XKRF (KZ)** **W/m-K**

Thermal conductivity of reflector for values of KZ not equal to KZPIN. **XKRF**(KZPIN) = thermal conductivity of cladding in the gas plenum region.

35 **DENSS**

Density of solid cladding at the reference temperature TR.

36 **COOLDN**

Not currently used.

37-43 **RHOCSI (KZ)** **J/m^3-K**

Density * heat capacity for the inner structure.

44-50 **RHOCSO (KZ)** **J/m^3-K**

Density * heat capacity for the outer structure.

51-57 **RHOCR (KZ)** **J/m^3-K**

Density * heat capacity for the reflector. For KZ = KZPIN (the pin zone), **RHOCR** is for the cladding in the gas plenum.

58 **RHOCG** J/m^3-K

Density * heat capacity for the gas in the gas plenum..

59 **RG** m^2-K/W

Thermal resistance of plenum gas, typical value = 0.06.

60 **FGMIN**

Fraction of retained fission gas that assists non-equilibrium fission gas bubble induced fuel swelling (burnup dependent).

61 **RST**

Not currently used.

62 **WRF**

Not currently used.

63 **RREF**

Not currently used.

64 **EMF**

Not currently used.

65 **EMS**

Not currently used.

66 **EMNA**

Not currently used.

67 **HMELT**

Not currently used.

68 **FCLOP**

Fraction of open porosity that must be removed before axial movement allowed with metal fuels $0 \leq$ `FCLOP` ≤ 1.

=0, Open porosity volume remains constant.

=1, All open porosity volume used before axial movement allowed.

69 **TSEP1** K

Fuel plasticity temperature = **TSEP1** + **TSEP2** * (fuel-cladding interface pressure - plenum gas pressure). The fuel pellet annulus at this temperature does not move radially inward or outward during calculations of fuel mesh boundaries.

Suggested value: 2700 for oxide fuel.

70 **TSEP2** K/Pa

Pressure adjustment to plasticity temperature.

Recommended value: 10^{-8}.

71 **BONDNA** kg

Mass of sodium added to fuel pin to produce the fuel-cladding bond.

72 **REFDEN** kg/m^3

Theoretical density of metal fuel at the reference temperature. Used for **IFUELM** = 2.

73 **FUELEX** 1/K

Fuel axial expansion coefficient. Used only with simple axial expansion feedback calculation. (See **IAXEXP**). Typical value: 1.1E-5.

74 **CLADEX** 1/K

Cladding axial expansion coefficient. Used only with simple axial expansion feedback calculation. (See **IAXEXP**). Typical value: 2E-5.

75 **YFUEL** Pa

Fuel young's modulus.

Typical value: 1.5E11.

Used with simple axial expansion and EBR-II feedback calculations.

76 **YCLAD** Pa

Cladding Young's modulus.

Typical value: 1.4E11.

Used with simple axial expansion and EBR-II feedback calculations.

77 **FULREX** 1/K

Fuel linear expansion coefficient for bond gap conductance calculation.

78 **CLDREX** 1/K

Cladding linear expansion coefficient for bond gap conductance calculation.

Note: **FULREX** and **CLDREX** are used only if **ISSFU2** = -1 and **IHGAP** = 0.

79 **EXPCFF**

Effective axial expansion multiplier, for simple axial expansion feedback model only.

80 **TTRANM** K

Transition temperature for determining the outer boundary of the middle fuel zone in the SSCOMP model for **IFUELO** = 1.

81 **TTRANC** K

Transition temperature for determining the outer boundary of the central fuel zone in the SSCOMP model for **IFUELO** = 1.

82-89 **HACHCH**

Channel-to-channel (duct wall-to-duct wall) heat transfer coefficient x area per unit height. Used only if **NCHCH** > 0.

Note: Use the total duct wall contact area for all subassemblies involved.

90 **POROSS**

Porosity of the outer zone of metal fuel at the end of steady-state irradiation. Used in the SSCOMP model for **IFUELO** = 1. See also **PORMSS** and **PORCSS**.

91 **PORMSS**

Porosity of the middle zone of metal fuel at the end of steady-state irradiation. Used in the SSCOMP model for **IFUELO** = 1. See also **POROSS** and **PORCSS**.

92 **PORCSS**

Porosity of the central zone of metal fuel at the end of steady-state irradiation. Used in the SSCOMP model for **IFUELO** = 1. See also **POROSS** and **PORMSS**.

93-103 **FPORNA**

Ratio of sodium filled porosity to total porosity in the metal fuel by radial mesh node at the end of the pre-transient irradiation. Used in SSCOMP for **IFUELO** = 1.

104 **TAUINV** 1/s

Inverse of heat transfer time constant. Degree of implicitness, THETA2, is:

THETA2 = (1 + X)/(2 + X)

X = DT * **TAUINV**

DT = time step size

Currently only used with the multiple pin option (**JJMLTP** not equal to 0).

Suggested value: 5.0 for metal fuel, 1.0 for oxide fuel.

FPIN-2 INPUT

105 **XEUTHR** **K**

Eutectic threshold temperature (K).

Default value = 988.

(See **FSPEC** for consistent input).

106 **XGBFRA**

Not currently used.

107-130 **XCLDHR (J)**

Pre-transient hardness parameter used in cladding flow stress calculation. Default value is 0.223, the value appropriate to 20% CW un-irradiated stainless steel.

131 **XFPLC0**

Fuel power law creep constant C0.

132 **XFPLC1**

Fuel power law creep constant C1.

133 **XFDVC0**

Fuel power law swelling constant C0.

134 **XFDVC1**

Fuel power law swelling constant C1.

135 **XCIPL0**

Cladding idealized flow stress constant C0.

136 **XCIPL1**

Cladding idealized flow stress constant C1.

137 **XHTERR**

Relative convergence criterion for heat transfer calculation. Default value = 0.0005.

138 **XEPSCA**

Relative convergence criterion for cavity pressure. Default value = 0.001.

139 **XEPSFE**

Relative convergence criterion for finite element analysis. Default value = 0.0005.

140 **XEPTES**

Relative convergence criterion for plastic creep strains. Default value = 0.0005.

141 **XEVTES**

Relative convergence criterion for swelling strains. Default value = 0.0005.

END FPIN-2 INPUT

142 **XMCFMC**

Reserved by FED.

143 **HACIFM**

Reserved by FED.

144 **HACOFM**

Reserved by FED.

145 **HAFMOP**

Reserved by FED.

146 **HAGPFM**

Reserved by FED.

147 **TOPLSS**

Reserved by FED.

148 **StructEX** **1/K**

Structure axial expansion coefficient. Used only with simple axial expansion feedback calculation. (See **IAXEXP**).

149 **CDOSECONV**

Conversion factor from clad fast neutron flux (n/cm^2/s/1.0E+22) to dose rate (dpa/s).

Only applicable if **IDEFOPT** > 0 and **ICTYPE** = 4 (15-15Ti cladding).

150 **COCOR** m

Clad outer corrosion thickness prior to transient initiation.

Only applicable if **IDEFOPT** > 0 and **ICTYPE** = 4 (15-15Ti cladding).

151 **CDFMEAN**

Mean value of the normal distribution, describing the predicted fraction of the fuel pins failed due to creep rupture as a function of Common Logarithm of CDF.

Only applicable if **IDEFOPT** > 0 and **ICTYPE** = 4 (15-15Ti cladding).

152 **CDFSIGMA**

Standard deviation value of the normal distribution describing the predicted fraction of the fuel pins failed due to creep rupture as a function of common logarithm of CDF.

Only applicable if **IDEFOPT** > 0 and **ICTYPE** = 4 (15-15Ti cladding).

153-200 **DUMPCH**

Not currently used.

2.7.15 Block 64 — COOLIN — Channel-Dependent Coolant Input

1 **AFR**

Liquid slug friction factor coefficients.
Default values when **IFRFAC** < 2: **AFR** = 0.1875, **BFR** = -0.2.
Required input when **IFRFAC** = 2.

2 **BFR**

see **AFR**. Required input when **IFRFAC** = 2.

3 **C1**

Coefficients in convection heat-transfer coefficient equation:
Hc = Kc*(**C1***(Re*Pr)****C2** + **C3**)/Dh
(Lyon-Martinelli correlation).

Kc: coolant thermal conductivity
Re: Reynolds number

Pr: Prandtl number
Dh: channel hydraulic diameter

4 C2

See **C1**

5 C3

See **C1**

6 DWMAX

Maximum fractional change per heat-transfer step in coolant flow rate before boiling.
Default: = 0.2

7 RELAM

Maximum Reynolds number for flow in the laminar regime.
Default value when **IFRFAC** < 2: **RELAM** = (**AFLAM**/**AFR**)**(1/(1 + **BFR**)) to ensure continuity at the turbulent/laminar boundary.
Required input when **IFRFAC** = 2.

8 AFLAM

Laminar friction factor = **AFLAM**/Re, where Re is the Reynolds number.
Suggested value: 64. Required input when **IFRFAC** = 2.

9 RETRAN

Minimum Reynolds number for flow in the turbulent regime. Required input when **IFRFAC** = 2.

10 C1TRAN

Transition regime friction factor coefficient. Default value: 1/3

11 C2TRAN

Transition regime friction factor coefficient. Default value: 7.0

12 C3TRAN

Transition regime friction factor coefficient. Default value: 1/3

13-46 DUM000

Not currently used.

47 **W0** **kg/s**

W, steady-state coolant flow rate per fuel-pin.

48-63 **XKORI (K,M)**

Orifice coefficients K < 8:

K = 1,2, …, **NREFB**+**NREFT**+1:
XKORI(K,M) is the coefficient at the bottom of zone K.

K = **NREFB**+**NREFT**+2:
XKORI(K,M) is the coefficient at the top of the subassembly.

M = 1, for upward flow.
M = 2, for downward flow.

Example: ((K=1,2,…),M=1,2)

64 **XKORGD**

Orifice coefficient for spacer grids in pin. See **NGRDSP**.

65 **DZIAB** **1/m**

Effective coolant inertial term below the subassembly inlet.
Suggested value: D/(2*ACC), where D is the subassembly inlet hydraulic diameter and ACC is the inlet cross-sectional flow area per pin.

66 **DZIAT** **1/m**

Effective coolant inertial term above the subassembly outlet.
Suggested value: D/(2*ACC), where D is the subassembly outlet hydraulic diameter and ACC is the outlet cross-sectional flow area per pin.

67 **THETA1**

= 0.5 normally; = 0.0 for implicit calculation.

68 **THETA2**

= 0.5 normally; = 1.0 for implicit calculation.

69 **DTLMAX** **K**

Maximum temperature change/coolant time step in liquid. Suggested value: 15.

70 **DTVMAX** **K**

Maximum temperature change/coolant time step in vapor. Suggested value: 50.

71 **DZIMAX** **m**

Maximum interface motion/coolant time step. Suggested value: 0.1.

72 **HCOND** **W/m^2-K**

Condensation heat-transfer conductance of coolant.

73 **SLMIN** **m**

Minimum initial liquid slug length. Recalculated as cladding melting is approached. Default: 0.02.

74 **TUPL** **K**

Temperature of liquid coolant re- entering the subassembly from above. Used only with PRIMAR-1.

75 **WFMIN** **m**

Minimum film thickness on cladding. Burnout is assumed for film thickness less than **WFMIN**. For static film option **WFMIN** is approximately 0.667*WF0. Film motion model is not operational. Default: 1.0E-7. (See **WFMIND** and **WFMNSD**).

76 **WFMINS** **m**

Minimum film thickness on structure. Default: **WFMIN**

77 **WFS00** **m**

Initial film thickness on structure. Default: WF0.

78-80 **FRACP**

Fraction of pins in this channel in failure group M. See **IGASRL**. (M)

81-83 **FRUPT**

Cladding life fraction for cladding failure in group M. See **IGASRL**. (M)

84 **WF0** **m**

Initial thickness of liquid coolant film left on the cladding in a voided region. Suggested value: That thickness corresponding to a total liquid volume fraction of 0.15 (spread between cladding and structure films) in the voided region.

85-104 **TFIS**

Not currently used. (I)

105-124 **PFIS**

Not currently used. (I)

125-144 **ZFISU**

Not currently used. (I)

145-162 **ZFISD**

Not currently used. (I)

163 **DTSWCH** K

Switch from pre-boiling to boiling when coolant temperature is within **DTSWCH** of boiling. Default: 5 K.

164 **DTSI** K

Superheat for bubble formation near a liquid-vapor interface.

165 **DTS** K

Superheat before any sodium vapor bubble can be formed.

166 **RESMLT**

Not currently used.

167 **TEMDLT**

Not currently used.

168 **AFRV**

Vapor friction factor = **AFRV***(Re)** **BFRV**.
Default values: **AFRV** = 0.316, **BFRV** = -0.25.

169 **BFRV**

see **AFRV**

170 **XMINL** m

Vapor pressure gradient calculation starts when vapor bubble length is **XMINL**. Default: 0.05.

171 **DTCMIN** s

Minimum coolant time step size before boiling.
Suggested value: 1.0E-6.

172 **WFMIND** m

Minimum cladding film thickness after **IFILM** film nodes have dried out.
Suggested value: 1.0E-7

173 **WFMNSD** m

Minimum structure film thickness after **IFILM** nodes have dried out.
Suggested value: 1.0E-7

174 **FVAPM**

Fraction of two-phase friction factor to be used in vapor calculation.

=0, For single-phase friction factor.
=1, For two-phase friction factor.

175 **AFRF**

Liquid film friction factor coefficient. Friction factor = **AFRF** * (Re) ** **BFRF**. Default: 0.316.

176 **BFRF**

Liquid film friction factor coefficient. Default: -0.25.

177 **TPDMIN** s

Minimum coolant time step size during boiling. Suggested value: 1.0E-6.

178 **DTACCL** s

Acceleration term is dropped from the momentum equation for Na vapor in a pressure drop bubble if the time step size is less than **DTACCL**. Suggested value: 1.0E-5.

179 **REFLAM**

Reynolds number for switch to laminar friction factor for liquid film.

180 **AFFLAM**

Laminar film friction factor = **AFFLAM**/Re, where Re is the Reynolds number.

181 **AGSRLS** m^2

Flow area between gas plenum and rupture point, used only if **IGASRL** = 1.

182 **GASKOR**

Orifice coefficient between gas plenum and rupture point, used only if **IGASRL** =1.

183 **PGRMIN** Pa

Shut off gas release when plenum pressure falls below **PGRMIN**, used only if **IGASRL** = 1.

184 **GASMW**

Molecular weight of plenum gas. Suggested value: 100-130.

185 **HEBOIL**

Boiling heat transfer enhancement factor. Suggested value: 1.0 for Na, 50-100 for D2O.

186-188 **TMFAIL** s

Pin failure time for pin group M for the gas release model. Used only if **IGASRL**=2. (M)

189 **UACH1**

Channel-to-channel heat flow per pin per unit height at axial node J from coolant in channel I to coolant in channel I+1 is calculated as:

190 **UACH2**

Q(J)=(**UACH1**(I)*K(J)+**UACH2**(I)*C(J)*(W(I)+W(I+1)))*(T(I,J)-T(I+1,J))

Where
T(I,J) = Coolant temperature
K(J) = Coolant thermal conductivity
C(J) = Coolant specific heat
W(I) = Coolant flow rate

Note: **UACH1** and **UACH2** are used only if **JJMLTP** is not equal to 0.

191 **REORFT**

Transition Reynolds number for the inlet orifice. The inlet orifice goes laminar for smaller Reynolds numbers. Used only if **JJMLTP** > 0.

192 **FLSWCH**

Switch to lumped node coolant temperature treatment when |W/W0| < **FLSWCH**. Only in reflector zones. Only in the pre-boiling calculation.

193 **RELAMV**

Vapor Reynolds number for switch from laminar to turbulent.

194 **AFLAMV**

Vapor laminar friction factor = **AFLAMV**/Re. | | |

195 **FLODTM** **s**

Film flooding time. Only used if **IFLOOD** = 1.

196 **QEMAX** **W/m^2**

Maximum heat flux from clad to vapor in a voided region. Default: no limit.

DETAILED COOLANT SUB-CHANNEL MODEL

197-200 **UACHM1 (K)**

Coolant sub-channel to sub-channel heat flow per pin per unit height at axial node J from channel I to channel L is calculated as:
Q(J)= (**UACHM1***k(J)+**UACHM2***C(J)*(W(I)+W(L+1))) *(T(I,J)-T(L,J))

where

T(I,J) = coolant temperature
k(J) = coolant thermal conductivity
C(J) = coolant heat capacity
W(I) = coolant flow rate
L = the Kth channel that channel I is in contact with

Note: UACH1 and UACH2 were used for a previous multiple pin model. **UACHM1**(K) and **UACHM2**(K) are used for the current detailed coolant sub-channel model.
Reserved for the detailed coolant sub-channel model.

201-204 **UACHM2 (K)**

See **UACHM1**.
Reserved for the detailed coolant sub-channel model.

205-208 **ALATRL (K)** **m**

ALATRL Coolant lateral flow area per pin per unit height.
Reserved for the detailed coolant sub-channel model.

209-212 **XKLAT (K)**

Lateral flow orifice coefficient.
Reserved for the detailed coolant sub-channel model.

213 **DPLTLM** **Pa**

Lateral flow goes laminar if pressure difference between adjacent sub-channels is less than **DPLTLM**.
Reserved for the detailed coolant sub-channel model.

214 **XKSWRL** **1/m**

Swirl flow between sub-channels = **XKSWRL***DZ*WAV

DZ = axial node height
WAV = average of sub-channel axial coolant flow rates
See also **KSWIRL**.
Reserved for the detailed coolant sub-channel model.

215 **DTNUL1** **s**

Time step for the single sub-assembly null transient.
Reserved for the detailed coolant sub-channel model.

216 **EPSFLW** **kg/s**

Iteration convergence criterion, flow rates
Reserved for the detailed coolant sub-channel model.

217 **EPSTMP** **K**

Iteration convergence criterion, coolant temperatures.
Reserved for the detailed coolant sub-channel model.

218 **EPSPRS** **Pa**

Iteration convergence criterion, coolant pressures.
Reserved for the detailed coolant sub-channel model.

219 **XCMPRS**

Compressibility multiplier for the mass conservation equation. Default = 1.0
Reserved for the detailed coolant sub-channel model.

220 **XLINRT** **m**

Inertial length for sub-channel to sub-channel flow.
Reserved for the detailed coolant sub-channel model.

221-300 **DUMCOO**

Not currently used.

2.7.16 Block 65 — FUELIN — Channel-Dependent Fuel Input

1 **FSPEC**

Fuel-pin failure criterion, threshold value. **IERSTP**

(See also **FMELTM** and **PCFAIL**).

See **MFAIL**.

2 **FMELTM**

The minimum fuel melt fraction that must exist at the failure node before PLUTO2 or LEVITATE is allowed to be called.

Suggested value: 0.2 (must be input > 0).

3 **DENS** **kg/m^3**

Liquid cladding density at the liquidus.

4 **AE** **1/K**

Coefficient of linear expansion of solid cladding (=1/3 volume coefficient).

5 **RHOCD** **1/K**

Volumetric coefficient of expansion for liquid cladding.

6 **VISMC** **Pa-s**

Viscosity of cladding at liquidus temperature. At higher temperatures,

viscosity= **VISMC***EXP(**AVISC** / **TELIQ** - **AVISC** /T).

7 **VISTR** **Pa-s**

Viscosity of cladding at solidus temperature. Between solidus and liquidus,

viscosity=(**VISTR**-**VISMC**)* (1-FMELTC)** **XVISC** + **VISMC**

where FMELTC is the melt fraction.

8 **VISSC** Pa-s

Viscosity of refrozen cladding. This is used as viscosity below solidus.

9 **XVISC**

Exponent in viscosity equation.

10 **CLADFR**

Moody friction factor for the molten cladding during turbulent cladding flow over the fuel rod.

11 **REBRK**

Laminar to turbulent transition Reynolds number for molten cladding flow.

12 **GMULTF**

Constants in molten cladding and sodium vapor two-phase friction multiplier.

Multiplier = CHOR*(1.0+**GMULTF***(1.0-MAX(ALPH, **ALPHCR**)))

where ALPH is void fraction. The incoherence (time delay) factor is:

CHOR=MAX(0.01,MIN(1.0,(FPS/ **FPS0**)** **EXPFPS**))

where FPS is the time in full-power seconds since cladding first moved in the channel.

13 **ALPHCR**

See **GMULTF**.

14 **FPS0**

See **GMULTF**.

15 **EXPFPS**

See **GMULTF**.

16 **AMELT**

Not currently used.

17 **CLSTHR**

Multiplier for frozen cladding-structure heat transfer.

Suggested value: 1.0.

18 **FRACPD**

Not currently used.

19 **PCFAIL** **Pa**

Pin cavity pressure at pin failure time. Only necessary if the transient DEFORM calculation is not used (**ISSFU2** .NE.1) or if CLAP preceded LEVITATE in this channel. May also be useful for overriding the DEFORM calculated cavity pressure for parametric studies. For **PCFAIL** < 1, the DEFORM calculated cavity pressure is used in PLUTO and LEVITATE. This is the recommended option when it can be used.

20 **AVISC** **K**

Constant in viscosity equation. (See **VISMC**).

21 **CPC** **J/kg-K**

Specific heat capacity of molten cladding at the liquidus temperature.

22 **FPINAC**

Areal fraction of fuel, with enthalpy exceeding the melting threshold of at least **FNMELT** of the heat of fusion, required to initiate PINACLE ; i.e. for in-pin pre-failure fuel motion to begin. See **FNMELT** and **CPINAC** .

Recommended value: 2 radial nodes.

23 **CPINAC**

Also, at least 3 contiguous axial nodes must satisfy the condition: Molten fuel fraction > **FPINAC** x **CPINAC** in order to initiate PINACLE. Recommended value: 0.5.

24 **RHOREF** **kg/m^3**

Reference density for metallic fuel at room temperature (273.15K). See also **FPRSTY**.

Used by MFUEL (**IFUELO** = 2).

25 **WUREF**

Fabricated uranium weight fraction in metallic alloy fuel when MFUEL is used (**IFUELO** = 2). Also reserved for the SSCOMP model (**IFUELO** = 1).

Internally, MFUEL will redefine **WUREF** = 1.0 - **WPUREF** - **WZRREF**

26 **WPUREF**

Fabricated plutonium weight fraction in metallic alloy fuel when MFUEL is used (**IFUELO** = 2). Also reserved for the SSCOMP model (**IFUELO** = 1).

27 **WZRREF**

Fabricated zirconium weight fraction in metallic alloy fuel when MFUEL is used (**IFUELO** = 2). Also reserved for the SSCOMP model (**IFUELO** = 1).

28 **FPRSTY**

As-fabricated volume fraction of fuel porosity. Fabricated fuel density is (1.0 - **FPRSTY**) * **RHOREF**.

Used by MFUEL (**IFUELO** = 2).

29 **RAPIDEUT**

Rapid eutectic scaling factor for MFUEL (**IFUELO** = 2). **RAPIDEUT** > 0.

Default = 1.0.

30 **HFMELT** **J/kg**

Fuel heat of fusion when MFUEL (**IFUELO** = 2) is used.

Default is determined from IFR Metallic Fuels Handbook data.

When not using MFUEL, see **UFMELT**.

31-51 **DFUELX**

Not currently used.

52 **FUSLDT** **m**

Maximum axial displacement of the upper fuel blanket or/and sodium slug above the fuel active top.

53 **FUSLMA** **kg/pin**

Total mass of the upper fuel blanket or/and sodium slug above fuel pin top.

54 **BURNFU** **at\%**

Axially averaged fuel pin burnup for metal fuel pins used in the SSCOMP model (**IFUELO** = 1). Also used for **IFUELM** = 2.

= 0, **FPDAYS** and **FLTPOW** used to calculate cladding fluence.
> 0, **BURNFU** determines cladding fluence.

55-198 **XPUZR**

Post-zone formation plutonium and Zirconium weight fractions in the central zone, followed by those in the middle zone and those in the outer zone.

Not currently used.

199 **GASRTM**

Not currently used.

200 **TIRRFU** **K**

Cladding steady state irradiation temperature at the top of the fuel region for metal fueled pins. Used in life fraction calculation.

≤ 0, Maximum temperature during steady state at each axial node used.
> 0, And **ISSFUE** = 0, value applied to each axial node

201 **SCDLA**

Sensitivity coefficient for lanthanide diffusion coeffcient in MFUEL.

Default value is 1.0. Must be greater than zero.

202 **SCDAC**

Sensitivity coefficient for actinide diffusion coeffcient in MFUEL.

Default value is 1.0. Must be greater than zero.

203 **SCEUR**

Sensitivity coefficient for empirical eutectic penetration rate in MFUEL.

Default value is 1.0. Must be greater than zero.

204 **SCCTC**

Sensitivity coefficient for cladding thermal creep strain rate. Applicable to all **ICTYPE** values used by MFUEL, OFUEL, and DEFORM-4.

= -1.0, No cladding thermal creep.
> 0.0, Cladding thermal creep strain rate sensitivity coefficient.

Default value is 1.0. Must be greater than zero or equal to -1.0.

205-300 **DFUELI**

Not currently used.

2.8 Example Problems

2.8.1 Protected Transient Overpower Example Problem

The protected transient overpower example problem, which is distributed with SAS4A/SASSYS-1 releases as "exam1", is an analysis of a protected reactivity insertion event in the EBR-II reactor. The basic SAS4A/SASSYS-1 input deck for this example problem was taken from Ref. 2-6. This input deck is the standard EBR-II SAS4A/SASSYS-1 model for the primary and secondary coolant loops. The core model in this deck is for a generic Mark-III reactor, and does not correspond to an actual or planned loading. The standard EBR-II plant protection system model documented in Ref. 2-7 was added to the input deck to provide scram functions, and an external, programmed reactivity insertion of 0.01 $/second to a maximum of 0.15 $ was assumed to drive the transient beginning from the steady-state conditions.

The input reactivity insertion raises the reactor power to 115% of nominal. At this power level, the plant protection system begins a reactor scram sequence by inserting a total of -3.7 $ of reactivity over 0.45 seconds, sharply reducing the power. Six seconds following the scram signal, a manual trip of both primary coolant pumps is assumed, followed six seconds later by a manual trip of the secondary pump. (The primary pump trip is not standard EBR-II operating practice, but is assumed here for the purpose of demonstrating SAS4A/SASSYS-1). The auxiliary pump continues to operate throughout the sequence.

The reactor power history and the flow history in channel 3, a high-power driver subassembly, are plotted in Figure 2.8.1. The power reduction begins at ten seconds, and the core flow reduction begins at approximately sixteen seconds. Over the next two minutes, the power drops to the decay heat level, and the flow seeks an equilibrium at the level provided by the auxiliary pump and natural circulation.

The peak fuel, cladding, and coolant temperature histories in channel 3 are plotted in Figure 2.8.2, along with the coolant saturation temperature. The fuel, cladding, and coolant temperatures rise with the power until the reactor scram. Temperatures then drop with the falling power, until the primary pumps are tripped and the flow reduction causes the temperatures to rise. The coolant saturation temperature falls as the coolant pressure drops during the coast-down of the two primary pumps. As the coolant temperature rises, the increased buoyancy causes the reactor coolant flow to increase slightly, until at about 45 seconds into the transient, a local temperature maximum occurs, at a level near the initial peak cladding temperature. From this time forward, temperatures slowly fall as the coolant flow adjusts to the combination of forced and natural circulation at decreasing decay heat power.

2.8.2 Unprotected Transient Overpower Example Problem

The unprotected transient overpower example problem, which is distributed with SAS4A/SASSYS-1 releases as "exam2", is an analysis of an unprotected (without scram) transient overpower (TOP) accident in a metal-fueled 3500 MWt pool-type reactor. This reactor has been analyzed previously for accident initiators that included unprotected loss-of-flow (LOF) and loss-of-heat-sink (LOHS) sequences as well as the transient overpower sequence [2-8]. However, the TOP sequence analyzed previously corresponded to removal of one control rod. For this analysis, the ramp reactivity addition rate was set to 0.10 $/s, and no limit was placed on the total reactivity added. This ramp rate results in an initial reactor power rise that approximately corresponds to the 8-second period employed in the TREAT M-Series tests [2-9], and the unlimited insertion assures that fuel melting and cladding failure will occur. Therefore, the whole-core sequence demonstrates 1) molten metal fuel behavior at conditions similar to those of the TREAT M-Series tests, 2) the ability of the SAS4A/SASSYS-1 metal fuel relocation modules, PINACLE and LEVITATE, to reproduce metal fuel performance observed in TREAT, and 3) the reactor safety implications of TREAT fuel relocation observations.

The TOP sequence assumed here is an unlimited 0.1 $/s reactivity ramp addition with failure of the plant protection and safety systems. It is also assumed that since no scram takes place, the reactor coolant pumps continue to

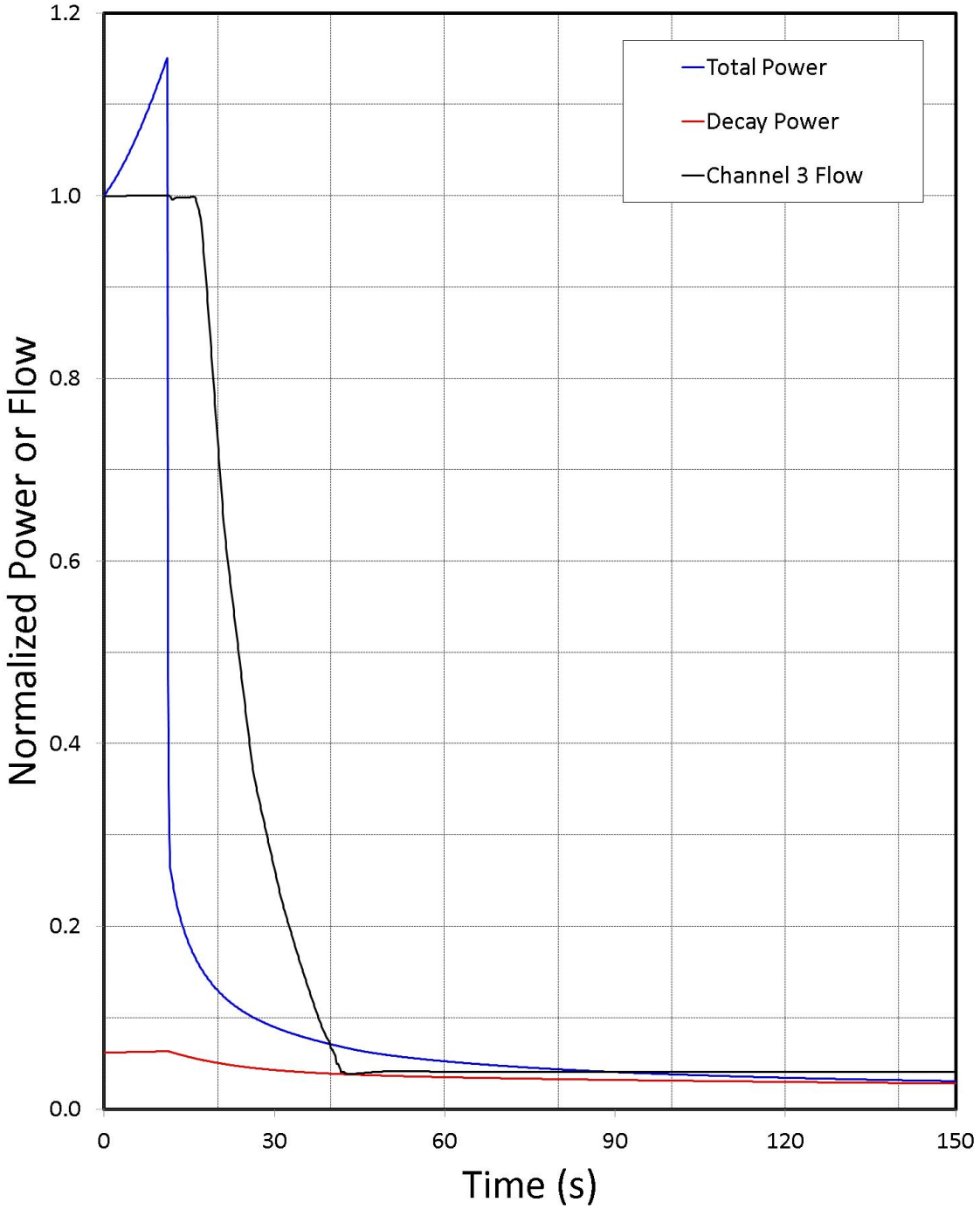


Figure 2.8.1: SAS4A/SASSYS-1 Reactor Power and Flow Histories

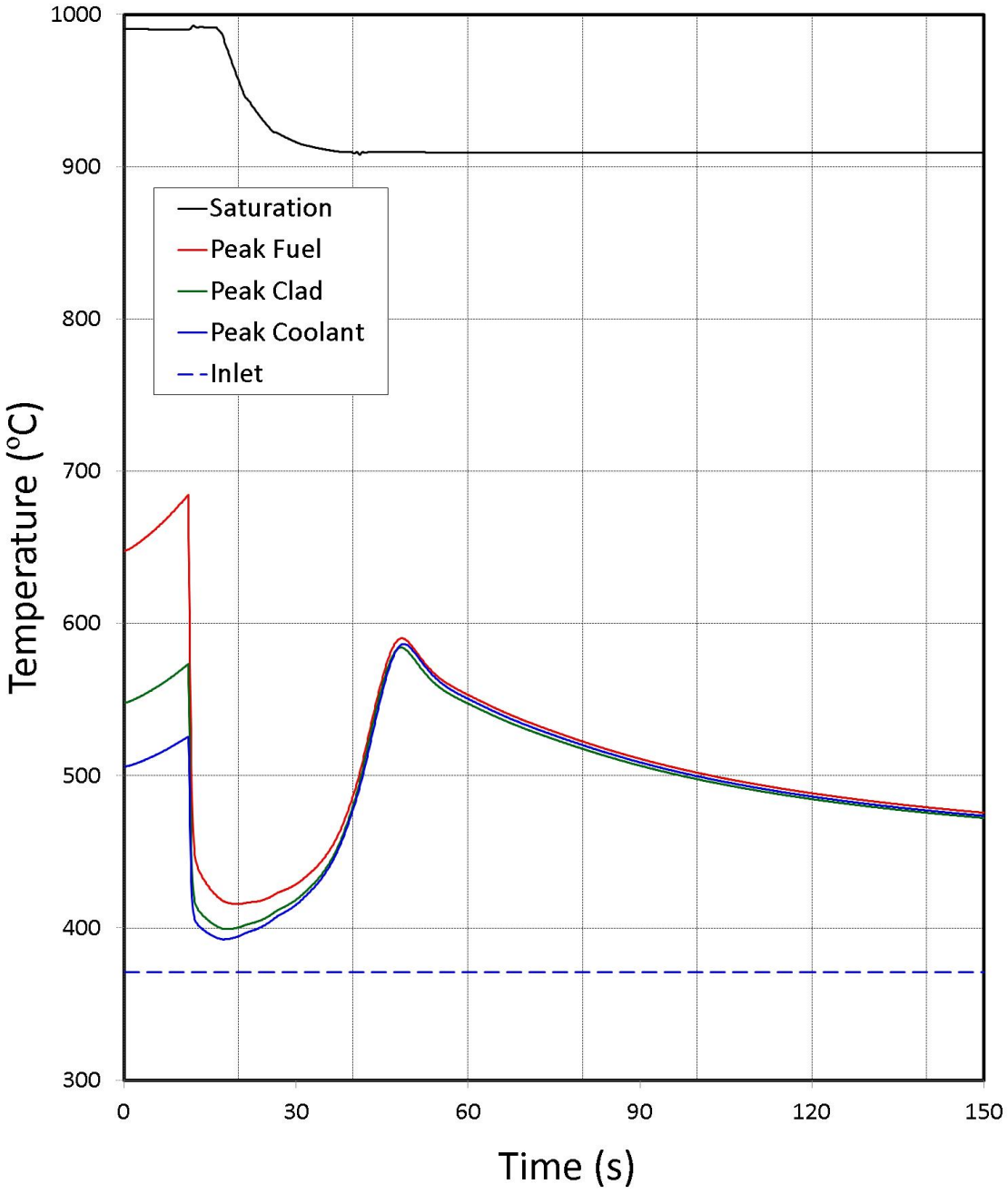


Figure 2.8.2: SAS4A/SASSYS-1 Temperature Histories in the High Temperature Driver Subassembly

operate. The reactor power history calculated by SAS4A/SASSYS-1 for this sequence is shown in Figure 2.8.3, and the corresponding net and component feedback reactivities are shown in Figure 2.8.4.

As shown in Figure 2.8.3, the reactor power rises initially in response to the ramp reactivity insertion, which is labeled as "Program" in Figure 2.8.4. The initial power ascension approximates the TREAT M-Series 8-second period. At about 7 seconds into the transient, conditions for in-pin fuel relocation are satisfied, and the PINACLE model begins execution to describe the expulsion of fuel within the cladding from the core into the fission gas plenum region, first for a few subassemblies and for more of the core as time goes on. This fuel relocation provides a negative reactivity effect that reduces the power temporarily, countering the input ramp reactivity.

2.9 References

2-1. G. L. Hofman, et al., Unpublished information, Argonne National Laboratory, 1985.

2-2. IFR Properties Evaluation Working Group, "Handbook of Properties of Metallic Fuels," Argonne National Laboratory, Idaho Falls, Idaho, June, 1988.

2-3. K. L. Derstine, Unpublished information, Argonne National Laboratory, 1984.

2-4. T. A. Taiwo and H. S. Khalil, "The DIF3D Nodal Kinetics Capability in Hex-Z Geometry - Formulation and Preliminary Tests," Proceedings of the International Topical Meeting Advances in Mathematics, Computations, and Reactor Physics, American Nuclear Society, Pittsburgh, Pennsylvania, April 28 - May 2, 1991.

2-5. T. H. Hughes and J. M. Kramer, "The FPIN2 Code - An Application of the Finite Element Method to the Analysis of the Transient Response of Oxide and Metal Fuel Elements," Conference on the Science and Technology of Fast Reactor Safety, British Nuclear Energy Society, Guernsey, May 12 - 16, 1986.

2-6. J. P. Herzog, Unpublished information, Argonne National Laboratory, 1993.

2-7. J. Y. Ku and T. C. Hung, Unpublished information, Argonne National Laboratory, 1992

2-8. R. A. Wigeland, R. B. Turski, and R. K. Lo, Unpublished information, Argonne National Laboratory, 1990.

2-9. T. H. Bauer, "Behavior of Modern Metallic Fuel in TREAT Transient Overpower Tests," *Nucl. Tech.*, vol. 92, p. 325, December, 1990.

2-10. "Function Parser for C++ v4.5.1," http://warp.povusers.org/FunctionParser

2-11. "Function Parser for C++ v4.5.1: Documentation," http://warp.povusers.org/FunctionParser/fparser.html#functionsyntax

2.10 Index of Input Variables

Input	Block	Location
AARIRV	PMR4IN	4410
AARORV	PMR4IN	4413
ABC	PMATCM	1085
ABCPU	PMATCM	1203
AC	PMATCM	1081
ACCHE	PMATCM	1104
ACCXE	PMATCM	1105

continues on next page

Table 2.10.1 – continued from previous page

Input	Block	Location
ACCZ	GEOMIN	1-7
ACLPEL	POWINA	448
ACLPRC	POWINA	421
ACRDEX	POWINA	73
ADJKTOL	OPCIN	155
ADOP	POWINC	62
AE	FUELIN	4
AFFLAM	COOLIN	180
AFLAM	COOLIN	8
AFLAMV	COOLIN	194
AFR	COOLIN	1
AFRF	COOLIN	175
AFRLRV	PMR4IN	4539
AFRTRV	PMR4IN	4538
AFRV	COOLIN	168
AGSRLS	COOLIN	181
AHBPAR	PMATCH	2
AHT	PMR4IN	3821-3824
AHX	PMR4IN	3841-3844
AIRAR2	PMR4IN	4420-4425
AIRARV	PMR4IN	4414-4419
AIRAUN	PMR4IN	4545
AIRTMP	PMR4IN	3933
AKCOND	PMATCM	1224
AKD	PMATCM	1272
ALATRL	COOLIN	205-208
ALFSS	PMATCM	1115
ALPHAP	PMR4IN	1499-1536
ALPHAT	PMR4IN	1537-1574
ALPHCR	FUELIN	13
AM	PMATCM	1119
AMELT	FUELIN	16
AMOTTK	PMR4IN	2223-2462
AMOTTK	PMR4IN	2223-2462
AMOTTK	PMR4IN	2223-2462
AMOTTK	PMR4IN	2223-2462
AMOTTK	PMR4IN	2223-2462
AMOTTK	PMR4IN	2223-2462
AMOTTK	PMR4IN	2223-2462
ANA	PMR4IN	3943
ANA	PMR4IN	4003
ANA	PMR4IN	4063
ANA	PMR4IN	4123
APG	PMATCM	1088
APMPHD	PMR4IN	1983-2222
APMPHD	PMR4IN	1983-2222
APMPHD	PMR4IN	1983-2222
APMPHD	PMR4IN	1983-2222
APMPHD	PMR4IN	1983-2222

continues on next page

Table 2.10.1 – continued from previous page

Input	Block	Location
APMPHD	PMR4IN	1983-2222
APMPHD	PMR4IN	1983-2222
APORE	PMATCM	1083
APROPI	PMATCM	1335-1394
AR	PMR4IN	3845-3848
AREACR	POWINA	446
AREAEL	PMR4IN	442-581
AREAIN	PMR4IN	1613-1650
AREAPC	GEOMIN	166
AREASG	PMR4IN	1721-1748
ASCRAM	OPCIN	23
ASI	PMR4IN	3849-3852
ASRALU	PMATCM	1280
ASTWL	PMR4IN	5027-5035
AVISC	FUELIN	20
AWALL	PMR4IN	2616-2653
AXHI	GEOMIN	8-31
AXMX	PMATCM	1165
AZEROX	PMATCM	1100
BCRDEX	POWINA	74
BDOP	POWINC	63
BENDNM	PMR4IN	862-1001
BETADK	POWINA	260-289
BETADN	POWINA	4-9
BETAHT	POWINA	320-324
BETSS	PMATCM	1116
BFR	COOLIN	2
BFRF	COOLIN	176
BFRTRV	PMR4IN	4542
BFRV	COOLIN	169
BHBPAR	PMATCH	3
BNDLOD	PMR4IN	1142
BNDMM1	POWINA	414
BNDMM2	POWINA	415
BONDNA	PMATCH	71
BTAPNA	PMR4IN	2464-2501
BTATNA	PMR4IN	2502-2539
BURNFU	FUELIN	54
C1	COOLIN	3
C1BY	PMR4IN	2694
C1BY2	PMR4IN	4300
C1BY3	PMR4IN	4303
C1BY4	PMR4IN	4306
C1IHX	PMR4IN	3653
C1IHXT	PMR4IN	4309
C1PIPE	PMR4IN	3650
C1RV	PMR4IN	4405
C1TRAN	COOLIN	10
C1VIPR	PMATCM	1155

continues on next page

Table 2.10.1 – continued from previous page

Input	Block	Location
C2	COOLIN	4
C2BY	PMR4IN	2695
C2BY2	PMR4IN	4301
C2BY3	PMR4IN	4304
C2BY4	PMR4IN	4307
C2IHX	PMR4IN	3654
C2IHXT	PMR4IN	4310
C2PIPE	PMR4IN	3651
C2RV	PMR4IN	4406
C2TRAN	COOLIN	11
C2VIPR	PMATCM	1156
C3	COOLIN	5
C3BY	PMR4IN	2696
C3BY2	PMR4IN	4302
C3BY3	PMR4IN	4305
C3BY4	PMR4IN	4308
C3IHX	PMR4IN	3655
C3IHXT	PMR4IN	4311
C3PIPE	PMR4IN	3652
C3RV	PMR4IN	4407
C3TRAN	COOLIN	12
C4IHX	PMR4IN	3656
C4IHXT	PMR4IN	4312
C4RV	PMR4IN	5325
CB2TC	POWINA	454
CDCL	PMATCM	1195
CDFMEAN	PMATCH	151
CDFSIGMA	PMATCH	152
CDFU	PMATCM	1140
CDNL	PMATCM	1142
CDOSECONV	PMATCH	149
CDVG	PMATCM	1144
CE	PMATCM	1070-1072
CFCOFV	PMATCM	1153
CFFURH	PMATCM	1154
CFLAT	GEOMIN	167
CFNACN	PMATCM	1146
CFNAEV	PMATCM	1147
CHBPAR	PMATCH	4
CIA1	PMATCM	1131
CIA2	PMATCM	1132
CIA3	PMATCM	1133
CIA4	PMATCM	1134
CIA5	PMATCM	1135
CIA6	PMATCM	1136
CIANDI	PMATCM	1208
CIANIN	PMATCM	1174
CIBBDI	PMATCM	1207
CIBBIN	PMATCM	1126

continues on next page

Table 2.10.1 – continued from previous page

Input	Block	Location
CIETFU	PMATCM	1143
CIFN	PMATCM	1137
CIFRFU	PMATCM	1128
CIFUFZ	PMATCM	1172
CIFUMO	PMATCM	1129
CIHCFU	PMATCM	1186
CINAF0	PMATCM	1125
CINAPN	PMATCM	1286
CIPINJ	PMATCM	1276
CIPNTP	PMATCM	1295
CIREFU	PMATCM	1127
CIRTFS	PMATCM	1170
CISP	PMATCM	1171
CIVIMT	PMATCM	1209
CIVOID	PMATCM	1130
CKD	PMATCM	1273
CKVFLO	PMR4IN	4619-4678
CKVORF	PMR4IN	4559-4618
CLADEX	PMATCH	74
CLADFR	FUELIN	10
CLADRA	POWINC	160-207
CLDREX	PMATCH	78
CLSTHR	FUELIN	17
CMFU	PMATCM	1188
CMNL	PMATCM	1141
CMWALL	PMR4IN	2654-2691
CNU	PMATCM	1117
COCOR	PMATCH	150
COEFDL	PMATCM	1200
COEFDL	PMATCM	1201
COEFDS	PMATCM	1
COEFDS	PMATCM	2
COEFDS	PMATCM	3
COEFK	PMATCM	4-10
COILP0	POWINC	375
COILP1	POWINC	376
COILP2	POWINC	377
COLFAC	PMATCM	1298
CONDCR	POWINA	442
COOLDN	PMATCH	36
CoolHeatTableID	INPMR4	1475
CPC	FUELIN	21
CPC0	PMR4IN	4680
CPC1	PMR4IN	4681
CPC2	PMR4IN	4682
CPC3	PMR4IN	4683
CPC4	PMR4IN	4684
CPCL	PMATCM	1196
CPCLRH	PMATCM	1197

continues on next page

Table 2.10.1 – continued from previous page

Input	Block	Location
CPCM	PMATCM	1287
CPCPMP	PMR4IN	4468-4497
CPCTAB	PMATCM	819-878
CPCTEM	PMATCM	879-898
CPFTAB	PMATCM	606-765
CPFTEM	PMATCM	766-785
CPFU	PMATCM	1139
CPINAC	FUELIN	23
CPMO	PMATCM	1288
CPSE	PMATCM	1212
CRDEXP	POWINA	72
CRDHA	POWINA	76
CRDLEN	POWINA	71
CRDMC	POWINA	75
CRMCI	PMR4IN	4516-4521
CRMCO	PMR4IN	4522-4527
CROETB	PMATCM	990-1049
CROETM	PMATCM	1050-1069
CRSAC	POWINA	426
CTLFNC	INCONT	4001-6000
CTLSIG	INCONT	6001-8000
CTLTAB	INCONT	1-2000
CTLTIM	INCONT	2001-4000
CUCM	PMATCM	1291
CUMO	PMATCM	1292
CUTOFF	PMR4IN	3963
CUTOFF	PMR4IN	4023
CUTOFF	PMR4IN	4083
CUTOFF	PMR4IN	4143
CVGMLT	PMR4IN	1749-1804
CVHE	PMATCM	1097
CVLMLT	PMR4IN	82-161
CVXE	PMATCM	1096
CZCM	PMATCM	1289
CZERO	PMATCM	1106
CZMO	PMATCM	1290
DABY	PMR4IN	2737-2744
DBBY	PMR4IN	2745-2752
DDBY	PMR4IN	2753-2760
DDX	PMATCM	1122
DDX2	PMATCM	1123
DECCON	POWINA	10-15
DEL	PMATCH	25
DELPUM	OPCIN	151
DELSCR	OPCIN	150
DENA	PMR4IN	3942
DENA	PMR4IN	4002
DENA	PMR4IN	4062
DENA	PMR4IN	4122

continues on next page

Table 2.10.1 – continued from previous page

Input	Block	Location
DENS	FUELIN	3
DENSS	PMATCH	35
DEW	PMR4IN	3940
DEW	PMR4IN	4000
DEW	PMR4IN	4060
DEW	PMR4IN	4120
DFLTCS	POWINA	419
DFLTSS	POWINA	420
DFUELI	FUELIN	205-300
DFUELX	FUELIN	31-51
DGO	PMATCH	26
DHAIRV	PMR4IN	4409
DHAORV	PMR4IN	4412
DHELEM	PMR4IN	582-721
DHPMP	PMR4IN	4288-4299
DHSEGG	PMR4IN	1805-1832
DHZ	GEOMIN	32-38
DKANSI	POWINA	801-880
DKBET2	POWINA	511-630
DKFRAC	POWINA	751-800
DKLAM	POWINA	290-319
DKLAM2	POWINA	631-750
DMP4I1	PMR4IN	3454
DMP4IN	PMR4IN	5328-5500
DNA	PMR4IN	3941
DNA	PMR4IN	4001
DNA	PMR4IN	4061
DNA	PMR4IN	4121
DPFCLD	PMR4IN	3962
DPFCLD	PMR4IN	4022
DPFCLD	PMR4IN	4082
DPFCLD	PMR4IN	4142
DPFHOT	PMR4IN	3965
DPFHOT	PMR4IN	4025
DPFHOT	PMR4IN	4085
DPFHOT	PMR4IN	4145
DPGRV0	PRIMIN	86
DPINMX	OPCIN	19
DPLTLM	COOLIN	213
DPRSTY	PMATCH	10
DPUO	PMATCM	1205
DPWMAX	OPCIN	22
DRCOLL	POWINA	425
DRFI	GEOMIN	189-195
DRFO	GEOMIN	169-175
DSHDRC	PMR4IN	3544-3547
DSHIHX	PMR4IN	3540-3543
DSTIZ	GEOMIN	39-45
DSTOZ	GEOMIN	46-52

continues on next page

Table 2.10.1 – continued from previous page

Input	Block	Location
DT0	OPCIN	5
DTACCL	COOLIN	178
DTCLAD	OPCIN	11
DTCMIN	COOLIN	171
DTDISR	PMATCM	1230
DTEVPF	PMR4IN	3441-3444
DTFAL1	PMATCM	1269
DTFAL2	PMATCM	1270
DTFAL3	PMATCM	1271
DTFUEL	OPCIN	10
DTIHX	PMR4IN	4284-4287
DTINMX	OPCIN	20
DTLMAX	COOLIN	69
DTMIN	OPCIN	95-104
DTMMAX	OPCIN	94
DTMMXB	OPCIN	21
DTMPTB	PMR4IN	2937-3104
DTMXB	OPCIN	6
DTNUL1	COOLIN	215
DTNULL	OPCIN	88
DTP0	OPCIN	13
DTPBOI	OPCIN	15
DTPFCI	OPCIN	17
DTPICL	OPCIN	93
DTPICP	OPCIN	92
DTPLEV	OPCIN	16
DTPLIN	PMATCM	1164
DTPLP	PMATCM	1168
DTPMAX	OPCIN	14
DTPNIN	PMATCM	1277
DTPNP	PMATCM	1279
DTS	COOLIN	165
DTSI	COOLIN	164
DTSIHX	PMR4IN	5192-5195
DTSSCP	OPCIN	26
DTSWCH	COOLIN	163
DTUDRC	PMR4IN	3552-3555
DTUIHX	PMR4IN	3548-3551
DTVMAX	COOLIN	70
DUM	PMR4IN	3992-3993
DUM	PMR4IN	4052-4053
DUM	PMR4IN	4112-4113
DUM	PMR4IN	4172-4173
DUM000	COOLIN	13-46
DUM002	POWINC	3
DUMCOO	COOLIN	221-300
DUMGEO	GEOMIN	285-350
DUMPCH	PMATCH	153-200
DUMPMC	PMATCM	1395-1725

continues on next page

Table 2.10.1 – continued from previous page

Input	Block	Location
DUMPNA	POWINA	969-1000
DUMPRI	PRIMIN	97-150
DWMAX	COOLIN	6
DXMAPR	OPCIN	28-87
DZBCGL	PRIMIN	90
DZBCGU	PRIMIN	91
DZEMFM	GEOMIN	275-284
DZIAB	COOLIN	65
DZIAT	COOLIN	66
DZIMAX	COOLIN	71
DZPLIN	PMATCM	1152
DZSHPX	PMR4IN	3455-3515
EFFIN	PMR4IN	5304-5307
EGBBLY	PMATCM	1161
EGFUSO	PMATCM	1151
EGMN	PMATCM	1178
EGSELQ	PMATCM	1211
EGSESO	PMATCM	1210
EL	PMR4IN	3939
EL	PMR4IN	3999
EL	PMR4IN	4059
EL	PMR4IN	4119
ELDRUM	PMR4IN	3961
ELDRUM	PMR4IN	4021
ELDRUM	PMR4IN	4081
ELDRUM	PMR4IN	4141
EllLossCoefTableID	INPMR4	1442
ELS	PMR4IN	3956
ELS	PMR4IN	4016
ELS	PMR4IN	4076
ELS	PMR4IN	4136
EMF	PMATCH	64
EMNA	PMATCH	66
EMS	PMATCH	65
EMSC	PMATCM	1109
EMSF	PMATCM	1108
ENPF	PMATCM	602
EPCH	PMATCM	1166
EPSCAV	PMR4IN	1971-1982
EPSCAV	PMR4IN	1971-1982
EPSCOM	PMATCM	1294
EPSCPL	INCONT	8002
EPSCS	INCONT	8001
EPSF	PMR4IN	4282
EPSFC	PMR4IN	4283
EPSFLW	COOLIN	216
EPSFS	PMR4IN	4534
EPSFSI	PMR4IN	4553-4558
EPSGAM	OPCIN	4

continues on next page

Table 2.10.1 – continued from previous page

Input	Block	Location
EPSGV	PMR4IN	4264-4269
EPSGVO	PMR4IN	4547-4552
EPSMS	PMATCM	1293
EPSOW	PMR4IN	4546
EPSPOW	OPCIN	3
EPSPRS	COOLIN	218
EPSREA	OPCIN	2
EPSRV	PMR4IN	4258-4263
EPSSCP	OPCIN	27
EPSSFP	PMATCM	1226
EPSTEM	OPCIN	1
EPSTMP	COOLIN	217
EPSTST	PMR4IN	5186-5188
ET	PMATCM	1103
EXKTB	PMATCM	11-70
EXKTM	PMATCM	71-90
EXPCFF	PMATCH	79
EXPCOF	PMATCM	1263
EXPFPS	FUELIN	15
EXSOTB	POWINA	471-490
EXSOTM	POWINA	491-510
FAXIAL	PMATCM	1258
FCDTCB	POWINA	453
FCDTR1	POWINA	422
FCDTR2	POWINA	423
FCDTRF	POWINA	424
FCLDWK	PMATCM	1259
FCLOP	PMATCH	68
FCR	POWINA	468
FDPTO	PMR4IN	3967
FDPTO	PMR4IN	4027
FDPTO	PMR4IN	4087
FDPTO	PMR4IN	4147
FFLAT	GEOMIN	168
FGCOND	PMATCM	1223
FGFI	PMATCM	1275
FGMIN	PMATCH	60
FGMM	PMATCM	600
FGPORX	PMATCM	1220
FGSPRD	PMATCM	1219
FHTCLD	POWINC	367
FHTFUL	POWINC	371
FIFNGB	PMATCM	1148
FINTHK	PMR4IN	5308-5311
FINVNT	PMR4IN	5324
FIRLIM	PMATCM	1266
FLODTM	COOLIN	195
FLOEXP	POWINA	447
FLOSSL	PMR4IN	2-41

continues on next page

Table 2.10.1 – continued from previous page

Input	Block	Location
FLOWBU	POWINC	267
FLSHRD	POWINA	445
FLSWCH	COOLIN	192
FLTPOW	POWINC	61
FMELTD	PMATCM	1260
FMELTM	FUELIN	2
FNARME	PMATCM	1176
FNDISR	PMATCM	1229
FNFUAN	PMATCM	1138
FNHTFU	PMATCM	1181
FNMELT	PMATCM	1169
FNSROS	PMATCM	1214
FNU	PMATCM	1118
FPDAYS	POWINC	1
FPINAC	FUELIN	22
FPKNG	POWINC	373
FPORNA	PMATCH	93-103
FPRSTY	FUELIN	28
FPS0	FUELIN	14
FRACP	COOLIN	78-80
FRACPD	FUELIN	18
FRFLOW	POWINA	70
FRMRSE	PMATCM	1213
FRPR	POWINA	69
FRUPT	COOLIN	81-83
FSMCI	PMR4IN	4504-4509
FSMCO	PMR4IN	4510-4515
FSPEC	FUELIN	1
FSTRAN	PMATCM	1261
FTMPCH	PMATCM	1262
FUELEX	PMATCH	73
FUELRA	POWINC	208-255
FULREX	PMATCH	77
FUNCOL	POWINC	372
FUNFLM	POWINC	370
FUNKCL	POWINC	368
FUNKFU	POWINC	369
FUNPOW	POWINC	374
FUSLDT	FUELIN	52
FUSLMA	FUELIN	53
FVAPM	COOLIN	174
FWPROF	PMR4IN	3974-3979
FWPROF	PMR4IN	4034-4039
FWPROF	PMR4IN	4094-4099
FWPROF	PMR4IN	4154-4159
FWTIME	PMR4IN	3968-3973
FWTIME	PMR4IN	4028-4033
FWTIME	PMR4IN	4088-4093
FWTIME	PMR4IN	4148-4153

continues on next page

Table 2.10.1 – continued from previous page

Input	Block	Location
G2PRDR	PMR4IN	1002-1141
GAMDBY	PMR4IN	2881-2904
GAMGAS	PMATCM	1299
GAMGS	PMATCM	1101
GAMGSC	PMR4IN	1689
GAMMA	PMATCM	1087
GAMNBY	PMR4IN	2857-2880
GAMSS	POWINC	2
GAMTNC	POWINC	4
GAMTNE	POWINC	5
GASKOR	COOLIN	182
GASMW	COOLIN	184
GASRTM	FUELIN	199
GATPF	PMATCM	601
GENTIM	POWINA	2
GK	PMATCM	1090
GK1	PMATCM	1092
GMULTF	FUELIN	12
GRAINK	PMATCM	1094
GRAINQ	PMATCM	1095
GRAVITY	OPCIN	154
GRAVTY	PMR4IN	2463
GSCRAM	OPCIN	25
GVMC	PMR4IN	4498-4503
GWO	PMR4IN	3952-3953
GWO	PMR4IN	4012-4013
GWO	PMR4IN	4072-4073
GWO	PMR4IN	4132-4133
HABYBY	PMR4IN	4184-4215
HACHCH	PMATCH	82-89
HACIFM	PMATCH	143
HACOFM	PMATCH	144
HAELHT	PMR4IN	4216-4245
HAFMOP	PMATCH	145
HAGPFM	PMATCH	146
HARDNS	PMATCM	1102
HAUIS	POWINA	406
HBMAX	PMATCH	5
HBMIN	PMATCH	6
HBPAR	PMATCH	7
HCCLMI	PMATCM	1187
HCFFMI	PMATCM	1179
HCFUBB	PMATCM	1180
HCOND	COOLIN	72
HCONRV	PMR4IN	4462-4467
HCSTW2	PMR4IN	5018-5026
HCSTWL	PMR4IN	5009-5017
HEADR	PMR4IN	1911-1922
HEADR	PMR4IN	1911-1922

continues on next page

Table 2.10.1 – continued from previous page

Input	Block	Location
HEADR	PMR4IN	1911-1922
HEADR	PMR4IN	1911-1922
HEADR	PMR4IN	1911-1922
HEBOIL	COOLIN	185
HECOND	PMATCM	1222
HEMASX	PMATCM	1227
HEMM	PMATCM	1225
HFBO	PMR4IN	3950
HFBO	PMR4IN	4010
HFBO	PMR4IN	4070
HFBO	PMR4IN	4130
HFIDRC	PMR4IN	3600-3603
HFIIHX	PMR4IN	3596-3599
HFILM	POWINA	444
HFMELT	FUELIN	30
HFPDRC	PMR4IN	3592-3595
HFPIHX	PMR4IN	3588-3591
HFSRV	PMR4IN	4456-4461
HFW	PMR4IN	3934
HFW	PMR4IN	3994
HFW	PMR4IN	4054
HFW	PMR4IN	4114
HGASRV	PMR4IN	4444-4449
HINVWL	PMR4IN	5036-5107
HITB	PMR4IN	3817-3820
HMELT	PMATCH	67
HOTB	PMR4IN	3813-3816
HSCO	PMR4IN	3949
HSCO	PMR4IN	4009
HSCO	PMR4IN	4069
HSCO	PMR4IN	4129
HSHO	PMR4IN	3951
HSHO	PMR4IN	4011
HSHO	PMR4IN	4071
HSHO	PMR4IN	4131
HSNK	PMATCH	8
HTCLGV	PMR4IN	5327
HTCPCR	POWINA	441
HWALL	PMR4IN	2578-2615
HWSHO	PMR4IN	3957
HWSHO	PMR4IN	4017
HWSHO	PMR4IN	4077
HWSHO	PMR4IN	4137
IADHX	INPMR4	1390-1393
IAREXT	INPCOM	40
IAXCON	INPCHN	284
IAXEXP	INPCHN	181
IAXTHF	INPCHN	124
IB3DM2	INPMR4	1144-1152

continues on next page

Table 2.10.1 – continued from previous page

Input	Block	Location
IBGO	INPCHN	79
IBINOT	INPMR4	893-971
IBINST	INPMR4	892
IBL3D2	INPMR4	498-506
IBL3DM	INPMR4	1479-1600
IBLPRN	INPCHN	119
IBLPRT	INPCOM	14
IBNEW	INPCHN	81
IBOP	INPCOM	70
IBOPLT	INPCOM	90
IBOWTP	INPCOM	102
IBSTOP	INPCHN	80
IBUBND	INPCHN	277
IBUGPL	INPCHN	75
IBUGPN	INPCHN	188
IBYBY	INPMR4	1020-1051
ICCVFS	INPMR4	1154
ICFDCV	INPMR4	1407-1410
ICFDDBG	INPMR4	1412
ICFINE	INPCHN	76
ICHCH	INPCHN	206-213
ICHCHT	INPCOM	79
ICHUIS	INPCHN	362
ICHUNK	INPCOM	47
iCIrrCreepID	INPCHN	235
ICLAD1	INPCOM	4
ICLADB	INPCHN	85
ICLADK	INPCHN	185
ICLADV	INPCHN	17
ICLCMP	INPCOM	24
ICLPRP	INPCOM	118
ICPDBG	INPMR4	498
ICRDDB	INPCOM	71
ICREXP	INPCOM	31
ICRNOD	INPCOM	75-77
ICRTMP	INPCOM	72-74
iCSwellID	INPCHN	234
ICTYPE	INPCHN	225
ICV2WL	INPMR4	1353-1361
ICVNAK	INPMR4	1378
ICVSSI	INPMR4	1380-1389
ICVSTR	INPMR4	1314-1316
ID2O	INPCOM	67
IDBDKH	INPCOM	85
IDBFLG	INPCHN	98-107
IDBGBL	INPCHN	120
IDBHTH	INPMR4	1142
IDBLST	INPCHN	121
IDBP4N	INPMR4	508

continues on next page

Table 2.10.1 – continued from previous page

Input	Block	Location
IDBPR4	INPMR4	507
IDBPWI	INPCOM	35
IDBRV	INPMR4	1290
IDBRVS	INPMR4	1291
IDBSTP	INPCHN	108-117
IDBSTR	INPMR4	1362
IDBUG0	INPCOM	2
IDBUGF	INPCHN	50
IDBUGV	INPCHN	1
IDCLDE	INPCHN	127
IDCLGO	INPCHN	125
IDCLSP	INPCHN	126
IDEFOPT	INPCHN	497
IDEFSTFAL	INPCHN	498
IDKCRV	INPCHN	203
IDKTYP	INPMR4	822-829
IDM51	INPCHN	198-202
IDMICH	INPCHN	499
IDMICH	INPCHN	546-600
IDNFLW	INPCOM	68
IDRCLC	INPMR4	493-496
IDRVACSKin	INPMR4	1440
IDRVACSTin	INPMR4	1439
IDRY	INPCHN	224
IDSSC	INPCHN	223
IELANE	INPMR4	1244-1273
IELBYP	INPMR4	830-837
IELDHX	INPMR4	992-995
IELDRP	INPMR4	477-480
IELDRS	INPMR4	485-488
IELHT	INPMR4	1052-1081
IELHT2	INPMR4	1082-1111
IELIHX	INPMR4	473-476
IELLCK	INPMR4	876-881
IELPMP	INPMR4	406-417
IELRVC	INPMR4	1276-1281
IELSGN	INPMR4	840-851
IELTPW	INPMR4	1367-1376
IELVLV	INPMR4	974-981
IEMGEM	INPCOM	104
IEMPMP	INPMR4	418-429
IEQMAS	INPCHN	118
IERSTP	INPCHN	2
IEUTOPT	INPCHN	197
IEVAP	INPMR4	864-875
IEXPFB	INPMR4	1009-1018
IFAE	INPCHN	184
IFAIL	INPCHN	87
IFCDHX	INPMR4	996-999

continues on next page

Table 2.10.1 – continued from previous page

Input	Block	Location
IFILM	INPCHN	128
IFIT	INPCOM	95-100
IFLOOD	INPCHN	494
IFLOW	INPCOM	20
IFMIOP	INPMR4	1311
IFPI01	INPCHN	286
IFPIN2	INPCHN	285
IFRFAC	INPCHN	186
IFSTEL	INPMR4	613-712
IFSTWL	INPMR4	1323-1325
IFT16	INPMR4	1365
IFT19	INPCOM	57
IFT1TM	INPCOM	119
IFT24	INPCHN	495
IFUEL1	INPCOM	3
IFUELB	INPCHN	16
IFUELC	INPCHN	193
IFUELI	INPCHN	132-155
IFUELM	INPCHN	192
IFUELO	INPCHN	191
IFUELV	INPCHN	15
IFUOPT	INPCHN	180
IFWC	INPMR4	1000-1003
IGASRL	INPCHN	278
IGHC	INPMR4	1004-1007
IGRLTM	INPCHN	279
IGSPRS	INPCHN	282
IHEALC	INPCHN	123
IHEX	INPCHN	29
IHGAP	INPCHN	24
IHTBYB	INPMR4	1193-1200
IHTBYD	INPMR4	1201-1208
IHTFLG	INPCHN	287
IHTPRS	INPCHN	96
IHXBYP	INPMR4	1172-1181
IHXCLC	INPMR4	489-492
iHXZID	INPMR4	1443-1446
ILAG	INPCHN	34
ILATF	INPCHN	496
ILGVHT	INPMR4	1477
ILIHXS	INPMR4	481-484
ILRPMP	INPMR4	430-469
ILSTEL	INPMR4	713-812
ILUBLK	INPCOM	48
iMassFractionTable	INPCHN	544
IMCVTY	INPCOM	34
IMELTV	INPCOM	28
IMETAL	INPCHN	189
IMKVPL	INPCHN	490

continues on next page

Table 2.10.1 – continued from previous page

Input	Block	Location
IMOMEN	INPCHN	182
INAKDR	INPMR4	509
INAPN	INPCOM	49
INAS3D	INPCOM	29
INDFAL	INPCHN	283
INEDIT	INPCOM	25
INPDU2	INPCOM	121-250
INRAEJ	INPCOM	44
INULLT	INPMR4	972
IOPCHC	INPCHN	226-233
IOPPL	INPCHN	276
IP4PRT	INPMR4	890
IPBDEN	INPCOM	106
IPGO	INPCHN	82
iPHXPRP	INPMR4	1415-1422
iPHXTID	INPMR4	1423-1430
iPHXWID	INPMR4	1431-1438
IPIC	INPCOM	101
IPINFG	INPCHN	486
IPINRE	INPCHN	487
IPIPTM	INPMR4	1310
IPL2A	INPMR4	1019
IPLOT	INPCHN	78
IPLTSG	INPCOM	69
IPLUP	INPCOM	5
IPMDFT	INPMR4	990
IPMPBY	INPMR4	1183-1192
IPNEW	INPCHN	84
IPNGO	INPCHN	194
IPNNEW	INPCHN	196
IPNPLT	INPCHN	190
IPNSTP	INPCHN	195
IPO	INPCOM	12
IPOBOI	INPCOM	13
IPORC	INPCHN	222
IPORFG	INPCHN	488
IPOWER	INPCOM	8
IPOWOP	INPCOM	9
IPOWRZ	INPCHN	364
IPRADJ	INPMR4	497
IPRD	INPCHN	97
IPRINT	INPCHN	77
IPRION	INPCOM	27
IPROPT	INPCOM	6
IPRSKP	INPCHN	274
IPRSNL	INPCOM	88
IPRSS1	INPCHN	489
IPSIG	INPCHN	95
IPSIZE	INPCHN	74

continues on next page

Table 2.10.1 – continued from previous page

Input	Block	Location
IPSTOP	INPCHN	83
IRAD	INPCHN	33
IRADEX	INPCOM	36
IRAPEN	INPCHN	280
IRATE	INPCHN	23
IRDEXP	INPCHN	187
IREACT	INPCOM	58
iReactivityTable	INPCHN	543
IREACZ	INPCHN	365
IRELAX	INPCHN	30
IRHOK	INPCHN	3
IROPT	INPCOM	65
IROR	INPCHN	45
IRVHTS	INPMR4	1478
IRVOPT	INPMR4	1275
ISCH	INPCOM	115
ISGCLC	INPMR4	852-863
ISGLNK	INPMR4	1377
iSGZID	INPMR4	1451-1462
ISIMPG	INPCOM	91
ISKDOT	INPCOM	116
ISLREA	INPMR4	1293-1300
ISNKRP	INPMR4	511
ISRCRP	INPMR4	510
ISSADX	INPMR4	1402-1405
ISSCPC	INPMR4	1288
ISSFU2	INPCHN	122
ISSFUE	INPCHN	32
ISSIHX	INPMR4	1155-1158
ISSNUL	INPCOM	87
ISSPMP	INPMR4	1159-1170
ISST15	INPMR4	1289
ISSTP	INPMR4	838
ISTDBS	INPMR4	1363
ISTGTB	INPMR4	1394-1397
ISTHTH	INPMR4	1143
ISTRVT	INPMR4	1317-1319
ISTSTP	INPMR4	1364
ISUBAS	INPCHN	89
ITABVV	INPMR4	982-989
ITARGE	INPCOM	103
ITAU	INPCHN	22
ITHPEN	INPMR4	1312
ITKEL	INPCOM	7
ITP20	INPCHN	204
ITREAT	INPCHN	275
ITYP24	INPCHN	522-541
ITYPCV	INPMR4	11-48
ITYPEL	INPMR4	49-188

continues on next page

Table 2.10.1 – continued from previous page

Input	Block	Location
IUM883	INPMR4	882-889
IVIS3D	INPCOM	120
IWLHRZ	INPMR4	1326-1334
IXSTPC	INPCOM	32
IXSTPF	INPCOM	33
IYLD	INPCOM	26
IZNC	INPCHN	366-389
IZNM	INPCHN	390-413
JCFT24	INPCHN	502-521
JCHMPN	INPCHN	215-218
JCLN	INPCHN	90
JCRIND	INPCOM	66
JCVG	INPMR4	269-324
JCVL	INPMR4	189-268
JFAIL	INPCHN	88
JFSELL	INPMR4	365-404
JJMLTP	INPCHN	214
JNCN	INPCHN	92
JNEN	INPCHN	91
JNSN	INPCHN	93
JPRNT1	INPCHN	46
JPRNT2	INPCHN	47
JREEXT	INPCOM	56
JRPRO	INPCHN	94
JRUPT	INPCHN	36
JSTRDX	INPCHN	183
KCHUIS	INPMR4	1209-1242
KDEBUG	INPCOM	82
KDENBU	INPCOM	94
KEBRS1	INPCOM	83
KEBRS2	INPCOM	84
KFAILP	INPCOM	37
KFIRR	INPCOM	93
KHDBK	INPCOM	92
KKSBRI	INPCHN	69
KKSBTP	INPCHN	68
KPROPI	INPCOM	114
KQSCRA	INPCOM	113
KSWIRL	INPCHN	220
KTING	INPCHN	19
KTRANC	INPCHN	236-249
KTRANM	INPCHN	250-273
KTREAT	INPCOM	112
KZEMFM	INPCHN	492
LBYP	INPMR4	1301
LCHTYP	INPCHN	281
LCPLAS	INPCHN	291
LCPROP	INPCHN	308
LCRACK	INPCHN	289

continues on next page

Table 2.10.1 – continued from previous page

Input	Block	Location
LCSWEL	INPCHN	293
LDBCPL	INPCHN	305
LDBFDV	INPCHN	304
LDBFPL	INPCHN	303
LDBOTA	INPCHN	311-334
LDBOTC	INPCHN	346-348
LDBOTF	INPCHN	335-345
LDBOUT	INPCHN	301
LDBSTP	INPCHN	302
LDETL	INPCHN	500
LELBYP	INPMR4	1302-1309
LFCSLP	INPCHN	295
LFPLAS	INPCHN	290
LFREQA	INPCHN	297
LFREQB	INPCHN	299
LFSWEL	INPCHN	292
LGAPCL	INPCHN	307
LGCLOS	INPCHN	310
LGPRES	INPCHN	306
LGRAPH	INPCHN	300
LHTOPT	INPCHN	288
LLRGST	INPCHN	294
LOUTSW	INPCHN	296
LQSLTP	INPCHN	363
LSKIPM	INPCHN	309
MAXSTP	INPCOM	11
MFAIL	INPCHN	86
MFREQA	INPCHN	298
MFTZN	INPCHN	414-485
MODEEX	INPCOM	55
MPL1	INPCHN	59
MPL2	INPCHN	60
MPL3	INPCHN	61
MPL4	INPCHN	62
MPL5	INPCHN	63
MPL6	INPCHN	64
MPL7	INPCHN	65
MPL8	INPCHN	66
MPL9	INPCHN	67
MSTEP	INPCHN	21
MSTPL1	INPCOM	109
MSTPL2	INPCOM	110
MSTPL3	INPCOM	111
MSTPLA	INPCOM	107
MSTPLB	INPCOM	108
MTACLP	INPCOM	53
MTCB	INPCOM	81
MTGRD	INPCOM	52
MTREAT	INPCHN	493

continues on next page

Table 2.10.1 – continued from previous page

Input	Block	Location
MTRFAC	INPCOM	63
MTRFT	INPCOM	64
MTRRAC	INPCOM	61
MTRRT	INPCOM	62
MTTLP	INPCOM	54
MULSTR	INPCOM	23
MZCHCH	INPCHN	491
MZLB	INPCHN	28
MZUB	INPCHN	27
NANEL	INPMR4	1243
NANRVC	INPMR4	1282-1287
NAXOP	INPCHN	20
NBINOT	INPMR4	891
NBYP	INPMR4	813
NCCV	INPMR4	1153
NCFDCV	INPMR4	1406
NCHAN	INPCOM	1
NCHCH	INPCHN	205
NCKV	INPMR4	472
NCLADM	INPCOM	30
NCPLEV	INPCOM	38
NCVD	INPMR4	3
NCVP	INPMR4	1
NCVS	INPMR4	2
NCVSSI	INPMR4	1379
NDELAY	INPCOM	16
NDHX	INPMR4	991
NDKGRP	INPCOM	17
NDRACS	INPMR4	471
NELEMT	INPMR4	10
NELHTN	INPMR4	1112-1141
NELML	INPMR4	325-364
NEXPFB	INPMR4	1008
NEXSO	INPCOM	105
NFT24	INPCHN	501
NFUELD	INPCOM	39
NGRAIN	INPCHN	31
NGRDSP	INPCHN	18
NIHX	INPMR4	470
NIHXBY	INPMR4	1171
NLINMX	INPCOM	117
NLNDWL	INPMR4	1344-1352
NNBUG1	INPCHN	48
NNBUG2	INPCHN	49
NODSUM	INPCHN	156-179
NOEQLE	INPCOM	80
NOEQPN	INPCOM	50
NOFDBK	INPCOM	89
NONE	INPCHN	349-360

continues on next page

Table 2.10.1 – continued from previous page

Input	Block	Location
NONE	INPMR4	1447-1462
NONE	INPMR4	1463-1474
NONE	PMR4IN	1899-1910
NONE	PMR4IN	1923-1982
NONE	PMR4IN	1899-1910
NONE	PMR4IN	1923-1934
NONE	PMR4IN	1959-1982
NONE	PMR4IN	1899-1982
NONEU0	INPCOM	21
NOREAC	INPCOM	41
NOSTRN	INPCHN	35
NPDKST	INPCOM	46
NPIN	INPCHN	25
NPK	INPCOM	10
NPLIN	INPCHN	37-44
NPLN	INPCHN	4
NPMPBY	INPMR4	1182
NPOWDK	INPCOM	45
NPREAT	INPCOM	18
NPRES	INPCOM	19
NPUMP	INPMR4	405
NREFB	INPCHN	5
NREFT	INPCHN	6
NROW	INPMR4	1398-1401
NRPI	INPCHN	70
NRPI1	INPCHN	71
NRPI2	INPCHN	72
NRPI3	INPCHN	73
NRREAC	INPMR4	1292
NRRNGS	INPCOM	60
NSCRVC	INPMR4	1274
NSEGCR	INPCOM	78
NSEGGD	INPMR4	9
NSEGGP	INPMR4	7
NSEGGS	INPMR4	8
NSEGLD	INPMR4	6
NSEGLP	INPMR4	4
NSEGLS	INPMR4	5
NSEGMP	INPCHN	361
NSGN	INPMR4	839
NSKIP	INPCHN	51-58
NSLEEX	INPCOM	42
NSRMTB	INPCOM	43
NSTEP	INPCOM	15
NSTRCV	INPMR4	1313
NSUBAS	INPCHN	26
NSUBTC	INPCOM	51
NSUBTR	INPCOM	59
NT	INPCHN	14

continues on next page

Table 2.10.1 – continued from previous page

Input	Block	Location
NT0TAB	INPCOM	22
NTGPT	INPMR4	512
NTHKPW	INPMR4	1366
NTLWBY	INPMR4	814-821
NTNODE	INPMR4	513-612
NULLCFD	INPMR4	1411
NULLD3	INPCOM	86
NULPT1	INPCHN	542
NULST1	INPCHN	221
NUMKLT	INPCHN	219
NUMWAL	INPMR4	1320-1322
NVALVE	INPMR4	973
NVNDWL	INPMR4	1335-1343
NZNODE	INPCHN	7-13
NZONF	INPCHN	129-131
OLDBDK	POWINA	16-21
OLDBDT	POWINA	28
OLDDKL	POWINA	22-27
OPCDUM	OPCIN	153
OPCDUM	OPCIN	156-250
ORFIN	PMR4IN	4540
P0GAS	PMATCH	27
PCFAIL	FUELIN	19
PCVSSI	PMR4IN	5294-5303
PD	PMR4IN	3958
PD	PMR4IN	4018
PD	PMR4IN	4078
PD	PMR4IN	4138
PDEC	PRIMIN	2
PDEC1	PRIMIN	3
PDEC2	PRIMIN	4
PERABY	PMR4IN	2921-2928
PERDBY	PMR4IN	2929-2936
PERFS	PMR4IN	4276-4281
PERFSI	PMR4IN	4432-4437
PERFSO	PMR4IN	4438-4443
PERGV	PMR4IN	4270-4275
PERGVO	PMR4IN	4426-4431
PERPTD	PMR4IN	3528-3531
PERPTX	PMR4IN	3524-3527
PERSPD	PMR4IN	3520-3523
PERSPX	PMR4IN	3516-3519
PERTID	PMR4IN	3536-3539
PERTIX	PMR4IN	3532-3535
PERVAC	PMR4IN	4252-4257
PERWL2	PMR4IN	4373-4402
PFIS	COOLIN	105-124
PGRMIN	COOLIN	183
PHPROF	PMR4IN	3986-3991

continues on next page

Table 2.10.1 – continued from previous page

Input	Block	Location
PHPROF	PMR4IN	4046-4051
PHPROF	PMR4IN	4106-4111
PHPROF	PMR4IN	4166-4171
PHPTI	PMR4IN	3959
PHPTI	PMR4IN	4019
PHPTI	PMR4IN	4079
PHPTI	PMR4IN	4139
PHTIME	PMR4IN	3980-3985
PHTIME	PMR4IN	4040-4045
PHTIME	PMR4IN	4100-4105
PHTIME	PMR4IN	4160-4165
PINMIN	PMR4IN	3453
PINSW	PMR4IN	3945
PINSW	PMR4IN	4005
PINSW	PMR4IN	4065
PINSW	PMR4IN	4125
PITCHA	POWINA	410
PITCHG	POWINA	409
PITCHT	POWINA	411
PLENL	GEOMIN	53
PLIN	POWINC	45-52
PLUT1	PMATCM	1248
PLUT10	PMATCM	1257
PLUT2	PMATCM	1249
PLUT3	PMATCM	1250
PLUT4	PMATCM	1251
PLUT5	PMATCM	1252
PLUT6	PMATCM	1253
PLUT7	PMATCM	1254
PLUT8	PMATCM	1255
PLUT9	PMATCM	1256
PMPEFR	PMR4IN	1947-1958
PMPEFR	PMR4IN	1947-1958
PMPEFR	PMR4IN	1947-1958
PMPEFR	PMR4IN	1947-1958
PMPFLR	PMR4IN	1935-1946
PMPFLR	PMR4IN	1935-1946
PMPFLR	PMR4IN	1935-1946
PMPFLR	PMR4IN	1935-1946
PMPHD	PMR4IN	3657-3719
PMPINR	PMR4IN	1899-1910
PMPINR	PMR4IN	1899-1910
PMPINR	PMR4IN	1899-1910
PMPSPR	PMR4IN	1923-1934
PMPSPR	PMR4IN	1923-1934
PMPSPR	PMR4IN	1923-1934
PMPTQ	PMR4IN	3720-3800
POD	PMR4IN	3944
POD	PMR4IN	4004

continues on next page

Table 2.10.1 – continued from previous page

Input	Block	Location
POD	PMR4IN	4064
POD	PMR4IN	4124
PORCSS	PMATCH	92
PORMSS	PMATCH	91
POROSS	PMATCH	90
POW	POWINA	1
POWLVL	POWINA	325-364
POWTIM	POWINA	365-404
POWTOT	POWINA	3
PPLCV	PMR4IN	4723-4760
PREATB	POWINA	29-48
PREATM	POWINA	49-68
PRESG0	PMR4IN	1461-1498
PRETAB	PRIMIN	5-24
PRETB2	POWINA	100-179
PRETM2	POWINA	180-259
PRETME	PRIMIN	25-44
PRFAIL	PMATCM	1177
PRFRA1	PMR4IN	2905-2912
PRFRA2	PMR4IN	2913-2920
PRSDRC	PMR4IN	3449-3452
PRSFTN	PMATCM	1297
PRSHAP	POWINC	256
PRSIHX	PMR4IN	3445-3448
PRSMIN	PMATCM	605
PRSTY	PMATCM	1073-1080
PRTDEL	PMATCM	1265
PRTSTR	PMATCM	1264
PSCRAM	OPCIN	24
PSHAPB	POWINC	319-366
PSHAPC	POWINC	271-318
PSHAPE	POWINC	6-29
PSHAPR	POWINC	30-44
PSHPBT	POWINC	262-266
PSHPBY	PMR4IN	2801-2856
PSHPTP	POWINC	257-261
PTCHRA	POWINA	450
PTCHRT	POWINA	451
PUBYU	POWINA	405
PUHALF	OPCIN	152
PUZRTP	PMATCM	1300-1315
PW	PMR4IN	3936
PW	PMR4IN	3996
PW	PMR4IN	4056
PW	PMR4IN	4116
PWLVL2	POWINA	881-920
PWO	PMR4IN	3946
PWO	PMR4IN	4006
PWO	PMR4IN	4066

continues on next page

Table 2.10.1 – continued from previous page

Input	Block	Location
PWO	PMR4IN	4126
PWTIM2	POWINA	921-960
PX	PRIMIN	1
QA1	PMATCM	1110
QA2	PMATCM	1111
QA3	PMATCM	1112
QA4	PMATCM	1113
QA5	PMATCM	1114
QEMAX	COOLIN	196
QETOT	POWINA	961-968
QFACT	PMR4IN	3954-3955
QFACT	PMR4IN	4014-4015
QFACT	PMR4IN	4074-4075
QFACT	PMR4IN	4134-4135
QKD	PMATCM	1274
QLAX	PMATCM	1120
QLAX2	PMATCM	1121
QPG	PMATCM	1089
QPORE	PMATCM	1084
QPU	PMATCM	1204
QSTAR	PMATCM	1202
QSWL	PMATCM	1082
QV	PMATCM	1091
QV1	PMATCM	1093
RAFPLA	PMATCM	1158
RAFPSM	PMATCM	1159
RAFUZ	GEOMIN	197-220
RALUDI	PMATCM	1284
RALUFZ	PMATCM	1285
RAPIDEUT	FUELIN	29
RBR	GEOMIN	54-77
RBR0	GEOMIN	180
RBRPL	GEOMIN	102
RCALBY	PMR4IN	2761-2768
RCAUBY	PMR4IN	2769-2776
RCBARR	POWINA	452
RCBLBY	PMR4IN	2777-2784
RCBUBY	PMR4IN	2785-2792
RCDBY	PMR4IN	2793-2800
RCORE	PMR4IN	5008
RCSHDR	PMR4IN	3560-3563
RCSHHX	PMR4IN	3556-3559
RCTUDR	PMR4IN	3568-3571
RCTUHX	PMR4IN	3564-3567
RDEXCF	POWINA	412
RDEXPC	POWINA	78
REAINS	POWINA	417
REAITR	OPCIN	8
REBRK	FUELIN	11

continues on next page

Table 2.10.1 – continued from previous page

Input	Block	Location
REDPTO	PMR4IN	3966
REDPTO	PMR4IN	4026
REDPTO	PMR4IN	4086
REDPTO	PMR4IN	4146
REFDEN	PMATCH	72
REFLAM	COOLIN	179
RELAM	COOLIN	7
RELAMV	COOLIN	193
REORFT	COOLIN	191
RER	GEOMIN	78-101
RER0	GEOMIN	181
RERPL	GEOMIN	103
RESMLT	COOLIN	166
RETRAN	COOLIN	9
REYTRV	PMR4IN	4537
RG	PMATCH	59
RGAS	PMATCM	1124
RGASC	PMR4IN	1690
RGASSI	PMATCM	1086
RGFV	PMATCM	1216
RGSV	PMATCM	1217
RHOCD	FUELIN	5
RHOCG	PMATCH	58
RHOCR	PMATCH	51-57
RHOCRD	POWINA	440
RHOCSI	PMATCH	37-43
RHOCSO	PMATCH	44-50
RHONAR	PMR4IN	2540-2577
RHOREF	FUELIN	24
RHOTAB	PMATCM	91-250
RHOTEM	PMATCM	251-410
RHOZN	PMATCM	1316-1323
RHOZRO	POWINA	470
RHSLBT	PMATCM	1198
RHSLTP	PMATCM	1199
RHSSLQ	PMATCM	1206
RHSSSO	PMATCM	1215
RINFP	GEOMIN	104-127
RITB	PMR4IN	3801-3804
RIZNC	GEOMIN	224-247
RIZNM	GEOMIN	248-271
RLEQ	PMATCM	603
RODID	POWINA	429
RODOD	POWINA	430
ROFC	PMATCM	1099
ROFF	PMATCM	1098
ROGSPI	PMATCM	1296
ROTB	PMR4IN	3805-3808
ROUGHG	PMR4IN	1833-1860

continues on next page

Table 2.10.1 – continued from previous page

Input	Block	Location
ROUGHL	PMR4IN	722-861
ROUTFP	GEOMIN	128-151
RR1TC	POWINA	427
RR2TC	POWINA	428
RREF	PMATCH	63
RST	PMATCH	61
RUEQ	PMATCM	604
RVHTAB	PMR4IN	4258-4269
RVHTMP	PMR4IN	4270-4281
RW5RV	PMR4IN	4536
SCCTC	FUELIN	204
SCDAC	FUELIN	202
SCDLA	FUELIN	201
SCEUR	FUELIN	203
SCRTAB	POWINA	80-89
SCRTME	POWINA	90-99
SECLIM	PMATCM	1267
SegLossCoefTableID	INPMR4	1441
SER	GEOMIN	182-188
SHRDID	POWINA	434-436
SHRDLN	POWINA	431-433
SHRDOD	POWINA	437-439
SIGSTB	PMR4IN	4535
SLA	PMR4IN	3964
SLA	PMR4IN	4024
SLA	PMR4IN	4084
SLA	PMR4IN	4144
SLANTD	PMR4IN	3608-3611
SLANTX	PMR4IN	3604-3607
SLLMAX	POWINA	408
SLMIN	COOLIN	73
SLRVC	PMR4IN	4450-4455
SPFIN	PMR4IN	5312-5315
SPLITF	PMR4IN	5326
SRFMLE	PMATCM	1231
SRFSTZ	GEOMIN	159-165
SRLEN	PMR4IN	3825-3828
STCOR	GEOMIN	176
STEBOL	PMATCM	1107
STRCRA	POWINC	378-425
StructEX	PMATCH	148
SUFU	PMATCM	1157
T0TAB	PRIMIN	45-64
T0TME	PRIMIN	65-84
TAIRVC	PMR4IN	4403
TAUGAS	PMR4IN	1861-1898
TAUINV	PMATCH	104
TAUPEN	PMR4IN	4977
TAUUP	PMR4IN	2693

continues on next page

Table 2.10.1 – continued from previous page

Input	Block	Location
TCLMAX	OPCIN	12
TCOSTP	OPCIN	9
TCVSSI	PMR4IN	5284-5293
TDTMIN	OPCIN	105-114
TECLMN	PMATCM	1184
TECLRL	PMATCM	1185
TEFAIL	PMATCM	1175
TELIQ	PMATCM	813-815
TEMDLT	COOLIN	167
TESOL	PMATCM	810-812
TFIS	COOLIN	85-104
TFLIQ	PMATCM	794-801
TFSOL	PMATCM	786-793
THETA1	COOLIN	67
THETA2	COOLIN	68
ThickWallTableID	INPMR4	1414
THKWAL	PMR4IN	4939-4976
THRLIM	PMATCM	1268
THT2CV	PMR4IN	4679
THT2VL	PRIMIN	96
TIFP	PMATCM	1173
TIMAX	OPCIN	7
TIMCNS	PMATCH	1
TIMDBG	PRIMIN	92
TIMMIX	PRIMIN	95
TINSRT	POWINA	416
TIPLMX	PMATCM	1167
TIPNMX	PMATCM	1278
TIRRFU	FUELIN	200
TLIMIT	POWINA	418
TLPEL	POWINA	449
TLPRRC	POWINA	413
TMAINS	OPCIN	115-149
TMDBP4	PMR4IN	1
TME	PMATCM	899-901
TMF	PMATCM	411-418
TMFAIL	COOLIN	186-188
TMIDFG	PMATCM	1218
TMO	PMR4IN	3947
TMO	PMR4IN	4007
TMO	PMR4IN	4067
TMO	PMR4IN	4127
TMPMTB	PMR4IN	3273-3440
TNAO	PMR4IN	3948
TNAO	PMR4IN	4008
TNAO	PMR4IN	4068
TNAO	PMR4IN	4128
TNAX	PMR4IN	3937
TNAX	PMR4IN	3997

continues on next page

Table 2.10.1 – continued from previous page

Input	Block	Location
TNAX	PMR4IN	4057
TNAX	PMR4IN	4117
TNTRY	PRIMIN	85
TOPLSS	PMATCH	147
TPCOEN	OPCIN	91
TPCOLE	OPCIN	90
TPCOST	OPCIN	89
TPDMIN	COOLIN	177
TPLCV	PMR4IN	4685-4722
TPLIN	POWINC	53-60
TPRFCI	OPCIN	18
TPWH	PMR4IN	5244-5283
TPWMC	PMR4IN	5204-5243
TR	PMATCM	419
TREFCV	PMR4IN	1651-1688
TRFU	PMR4IN	1692
TRKLSC	PMR4IN	1959-1970
TRKLSC	PMR4IN	1959-1970
TRKLSC	PMR4IN	1959-1982
TSEP1	PMATCH	69
TSEP2	PMATCH	70
TSNK	PMATCH	9
TTRANC	PMATCH	81
TTRANM	PMATCH	80
TUPL	COOLIN	74
TW6RV	PMR4IN	4528-4533
TWASTI	GEOMIN	272
TWASTO	GEOMIN	273
U0CVGS	PMR4IN	1691
UACH1	COOLIN	189
UACH2	COOLIN	190
UACHM1	COOLIN	197-200
UACHM2	COOLIN	201-204
UEMELT	PMATCM	816-818
UFMELT	PMATCM	802-809
UIVOL	POWINA	77
UMELT	PMATCM	902-909
UN1281	PMATCM	1281
UN1282	PMATCM	1282
UN1283	PMATCM	1283
UNFSMC	PMR4IN	4544
UNGVMC	PMR4IN	4543
VFC	GEOMIN	196
VFCRD	POWINA	443
VFLREF	GEOMIN	178
VFNALQ	PMATCM	1160
VFNARE	PMATCM	1163
VFPLNT	GEOMIN	177
VFUREF	GEOMIN	179

continues on next page

Table 2.10.1 – continued from previous page

Input	Block	Location
VIFI	PMATCM	1145
VIFULQ	PMATCM	1162
VINL	PMATCM	1149
VISMC	FUELIN	6
VISSC	FUELIN	8
VISTR	FUELIN	7
VIVG	PMATCM	1150
VOIDRA	POWINC	112-159
VOLBLI	PMR4IN	5183-5185
VOLD	PMR4IN	3960
VOLD	PMR4IN	4020
VOLD	PMR4IN	4080
VOLD	PMR4IN	4140
VOLGS0	PMR4IN	3612-3649
VOLLGC	PMR4IN	1423-1460
VOLMIX	PMR4IN	2692
VSIHX1	PMR4IN	5196-5199
VSIHX2	PMR4IN	5200-5203
VSLEXP	PMR4IN	4174-4183
W0	COOLIN	47
WALLH	PMR4IN	1283-1422
WALLH2	PMR4IN	4343-4372
WallHeatTableID	INPMR4	1476
WALLMC	PMR4IN	1143-1282
WALMC2	PMR4IN	4313-4342
WALTH2	PMR4IN	4978-5007
WALTHK	PMR4IN	4799-4938
WDOPA	POWINC	64-111
WEVI	PMR4IN	3935
WEVI	PMR4IN	3995
WEVI	PMR4IN	4055
WEVI	PMR4IN	4115
WF0	COOLIN	84
WFMIN	COOLIN	75
WFMIND	COOLIN	172
WFMINS	COOLIN	76
WFMNSD	COOLIN	173
WFS00	COOLIN	77
WPUREF	FUELIN	26
WRF	PMATCH	62
WST	PMATCM	1221
WUREF	FUELIN	25
WZRREF	FUELIN	27
XCIPL0	PMATCH	135
XCIPL1	PMATCH	136
XCLDHR	PMATCH	107-130
XCMPRS	COOLIN	219
XEPSCA	PMATCH	138
XEPSFE	PMATCH	139

continues on next page

Table 2.10.1 – continued from previous page

Input	Block	Location
XEPTES	PMATCH	140
XEUTHR	PMATCH	105
XEVO	PMR4IN	3938
XEVO	PMR4IN	3998
XEVO	PMR4IN	4058
XEVO	PMR4IN	4118
XEVTES	PMATCH	141
XFDVC0	PMATCH	133
XFDVC1	PMATCH	134
XFPLC0	PMATCH	131
XFPLC1	PMATCH	132
XGBFRA	PMATCH	106
XHTERR	PMATCH	137
XKALBY	PMR4IN	2697-2704
XKAUBY	PMR4IN	2705-2712
XKBLBY	PMR4IN	2713-2720
XKBUBY	PMR4IN	2721-2728
XKDBY	PMR4IN	2729-2736
XKHXLS	PMR4IN	3829-3832
XKLAT	COOLIN	209-212
XKORGD	COOLIN	64
XKORI	COOLIN	48-63
XKRF	PMATCH	28-34
XKRLS	PMR4IN	3833-3836
XKSHDR	PMR4IN	3576-3579
XKSHHX	PMR4IN	3572-3575
XKSILS	PMR4IN	3853-3892
XKSITM	PMR4IN	3893-3932
XKSOLS	PMR4IN	3837-3840
XKSTIZ	PMATCH	11-17
XKSTOZ	PMATCH	18-24
XKSWRL	COOLIN	214
XKTAB	PMATCM	420-579
XKTB	PMR4IN	3809-3812
XKTEM	PMATCM	580-599
XKTUDR	PMR4IN	3584-3587
XKTUHX	PMR4IN	3580-3583
XLAIRV	PMR4IN	4408
XLAORV	PMR4IN	4411
XLE10	PMATCM	1238
XLE4	PMATCM	1232
XLE5	PMATCM	1233
XLE6	PMATCM	1234
XLE7	PMATCM	1235
XLE8	PMATCM	1236
XLE9	PMATCM	1237
XLENEL	PMR4IN	302-441
XLENGG	PMR4IN	1693-1720
XLENTR	PMR4IN	5189-5191

continues on next page

Table 2.10.1 – continued from previous page

Input	Block	Location
XLEPT1	PMATCM	1239
XLEPT2	PMATCM	1240
XLEPT3	PMATCM	1241
XLEPT4	PMATCM	1242
XLEPT5	PMATCM	1243
XLEPT6	PMATCM	1244
XLEPT7	PMATCM	1245
XLEPT8	PMATCM	1246
XLEPT9	PMATCM	1247
XLINRT	COOLIN	220
XLOGNA	PMATCM	1324-1331
XLP8	PMATCM	1190
XLP9	PMATCM	1191
XLRVC	PMR4IN	4246-4251
XLTUBE	PMR4IN	5316-5319
XLUNRV	PMR4IN	4541
XMCFMC	PMATCH	142
XMCSTW	PMR4IN	5108-5179
XMCUI	POWINA	407
XMCXAC	POWINA	79
XMINL	COOLIN	170
XMXMSI	PRIMIN	93
XMXMSO	PRIMIN	94
XPL10	PMATCM	1192
XPL11	PMATCM	1193
XPL12	PMATCM	1194
XPL5	PMATCM	1182
XPL6	PMATCM	1183
XPL7	PMATCM	1189
XPUZR	FUELIN	55-198
XRFSHP	POWINC	268
XRNSHP	POWINC	269
XRSSHP	POWINC	270
XSIGMC	PMATCM	1332
XSIGMD	PMATCM	1334
XSIGMK	PMATCM	1333
XTTUBE	PMR4IN	5320-5323
XVISC	FUELIN	9
XXNPIN	GEOMIN	274
YABOW	POWINA	466
YBBOW	POWINA	467
YCLAD	PMATCH	76
YDELT0	POWINA	465
YFUEL	PMATCH	75
YKHF	POWINA	456
YKNF	POWINA	455
YKNNA	POWINA	457
YKNSS	POWINA	458
YLCLR	POWINA	460

continues on next page

Table 2.10.1 – continued from previous page

Input	Block	Location
YLDTAB	PMATCM	910-969
YLDTEM	PMATCM	970-989
YRCCR	POWINA	462
YRCDOP	POWINA	464
YRCGP	POWINA	463
YRCRR	POWINA	461
YRCUR	POWINA	459
YTCUT	POWINA	469
ZBRVC	PMR4IN	4404
ZCENTR	PMR4IN	3105-3272
ZCHOBT	GEOMIN	221
ZCHOTP	GEOMIN	222
ZCVL	PMR4IN	1575-1612
ZFISD	COOLIN	145-162
ZFISU	COOLIN	125-144
ZIHX	PRIMIN	89
ZINL	PMR4IN	42-81
ZINST	PMR4IN	5180-5182
ZOFFST	GEOMIN	223
ZONEL	GEOMIN	152-158
ZoneLossCoefTableID	INPCHN	545
ZOUTEL	PMR4IN	162-301
ZPLENC	PMR4IN	4761-4798
ZPLENL	PRIMIN	87
ZPLENU	PRIMIN	88
ZSWFAC	PMATCM	1228

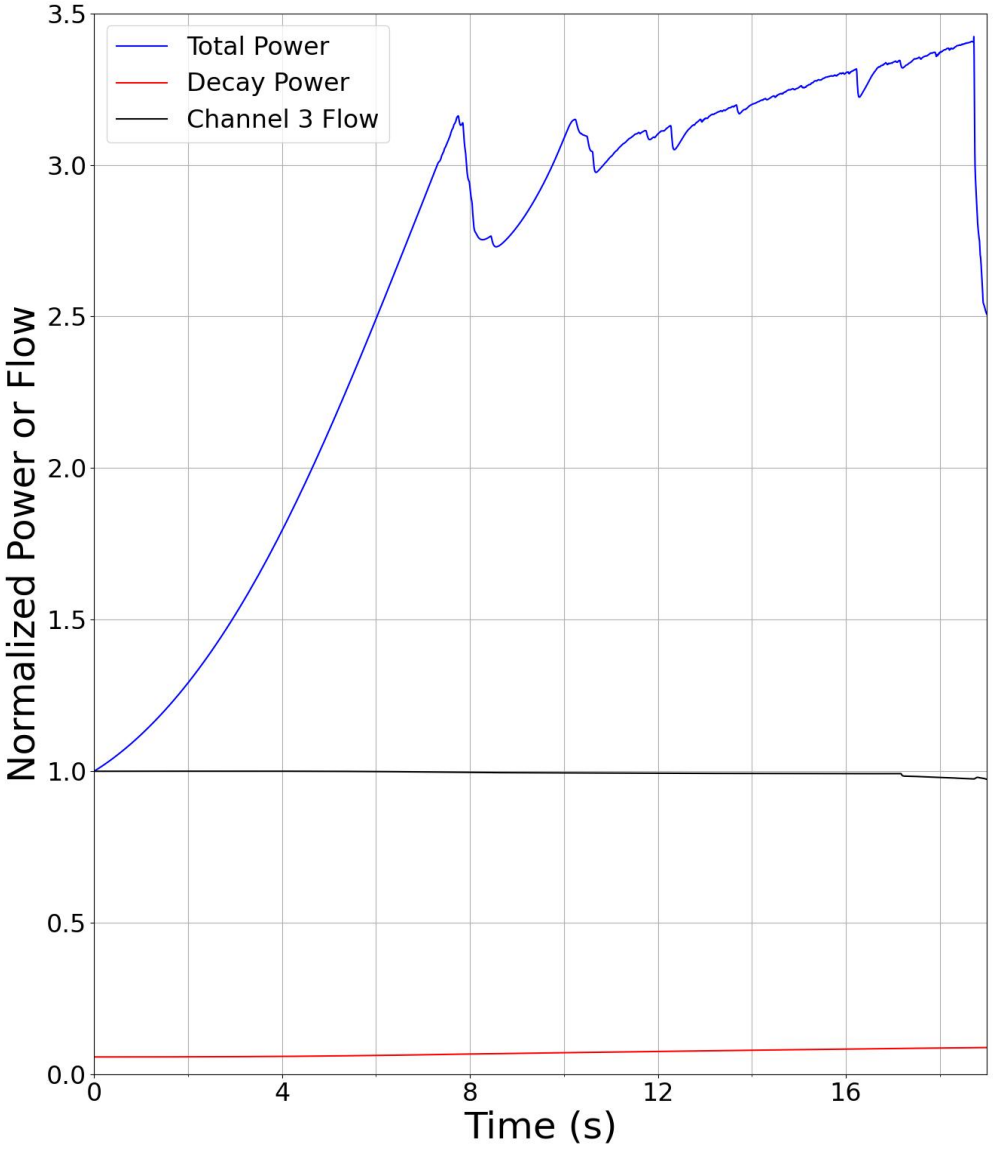


Figure 2.8.3: SAS4A/SASSYS-1 Example Problem Reactor Power History

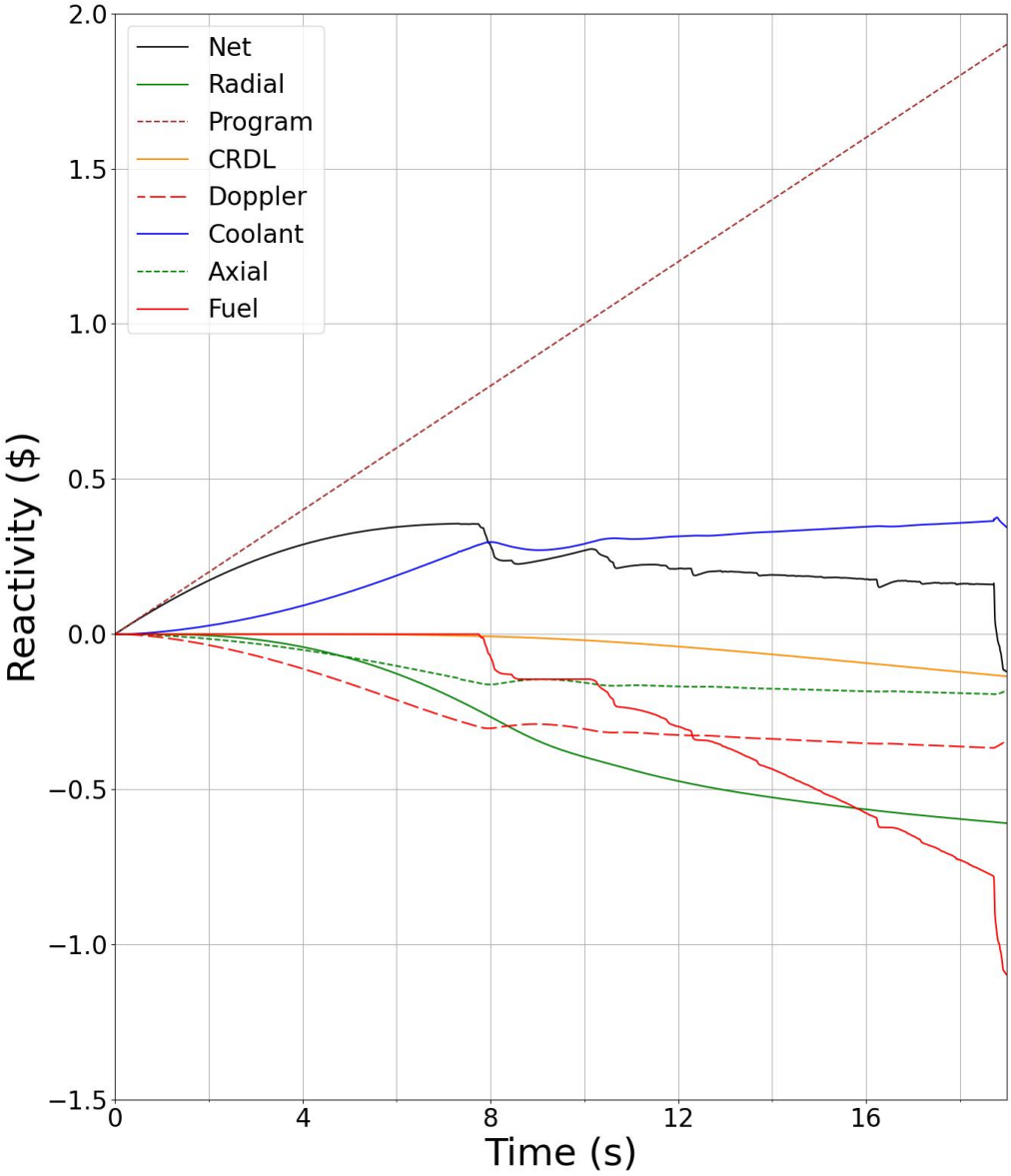


Figure 2.8.4: SAS4A/SASSYS-1 Example Problem Reactivity History

Part II: Whole-Plant Analysis

CHAPTER

THREE

CORE THERMAL-HYDRAULICS

3.1 Introduction

The core assembly thermal hydraulics treatment in SAS4A/SASSYS-1 includes the calculation of fuel, cladding, coolant, and structure temperatures, as well as coolant flow rates and pressure distributions. This treatment includes melting of the fuel and cladding. Boiling of the coolant is also handled, as described in Chapter 12. The relocation of fuel and cladding after pin disruption is described in Chapter 13, Chapter 14, and Chapter 16; and relocation of molten fuel before pin disruption is described in Chapter 15.

Prior to version 3.0, all of the core subassembly models in SAS4A/SASSYS-1 were single pin models: a single fuel pin and its associated coolant were used to represent a subassembly; and pin-to-pin variations within a subassembly were ignored. A multiple pin option has been added to the code in version 3.0. A number of pins and their associated coolant can now be used to represent a subassembly, so variations within a subassembly can be accounted for. Currently the multiple pin option is only available for single-phase thermal hydraulics; it does not handle coolant boiling, in-pin fuel relocation, or pin disruption. Therefore, typical SAS4A/SASSYS-1 cases that do not get into coolant boiling can be handled with the multiple-pin model, but typical core disruption cases can only be handled with single pin models.

Although SAS4A/SASSYS-1 is mainly a transient code, both steady-state and transient temperatures and coolant pressures are calculated. The steady-state solutions are obtained from the transient equations after dropping all time derivatives. In general, the steady-state solutions in the single pin per subassembly model are not obtained by running a transient calculation at constant power and flow until the results approach a steady-state solution. Instead, the steady-state temperatures are obtained rapidly from a direct solution based on conservation of energy and the use of the same spatial finite differencing as used in the transient. On the other hand, a direct steady-state solution for the multiple pin option would be much more complicated, especially if subassembly-to-subassembly heat transfer is included. Therefore, a null transient with powers and flows held constant is used to obtain steady-state conditions for the multiple pin option.

The thermal hydraulics calculations are carried out in a number of separate modules, and each module is designed for a specific type of calculation. A steady-state thermal hydraulics module provides the initial conditions for the transient. The transient temperatures are calculated in a pre-voiding module (TSHTRN) until the onset of boiling. After the onset of boiling, the fuel-pin temperatures are calculated in a separate module (TSHTRV) that couples with the boiling module.

The core thermal-hydraulic routines interact with a number of other modules, as shown in Figure 3.1.1. Before the onset of voiding, TSHTRN calculates the coolant temperatures used in the hydraulic calculations, whereas the hydraulics routines calculate the coolant flow rates used in TSHTRN. After the onset of voiding, coolant temperatures are calculated in TSBOIL, and this module supplies the heat flux at the cladding outer surface or the fuel outer surface to TSHTRV. TSBOIL uses the cladding temperatures from TSHTRV in its coolant temperature calculations. The point kinetics module supplies the power level used in the heat-transfer routines, and the heat-transfer routines supply the Doppler feedback reactivity as well as other temperature-dependent reactiv-

ity feedback. TSBOIL supplies the voiding reactivity. The inlet plenum temperature computed by PRIMAR-4 is used in calculating the inlet temperature for TSHTRN or TSBOIL, and TSHTRN or TSBOIL provides the subassembly outlet temperatures used by PRIMAR-4 to compute the outlet plenum temperature. If flow reversal occurs in a subassembly, then the outlet plenum temperature computed by PRIMAR-4 is used in calculating the coolant temperature at the top of the subassembly, and the temperature computed by TSHTRN or TSBOIL for the coolant leaving the bottom of the subassembly is used by PRIMAR-4 to calculate the inlet plenum temperature. PRIMAR-4 supplies the inlet and outlet plenum pressures that drive the coolant hydraulics calculations, and the core channel flows are provided to PRIMAR-4 by TSBOIL and the pre-voiding hydraulics. The initial coolant flow rate and pressure distribution are supplied to TSBOIL by the pre-voiding hydraulics routines.

The transient calculations in the code used a multi-level time step approach, with separate time steps for each module. For the heat-transfer routines, all temperatures are known at the beginning of a heat-transfer step, and the routines calculate the new temperatures at the end of the step. The heat-transfer time step can be longer than the coolant time step or the PRIMAR time step, but the heat-transfer time step can be no longer than the main time step that is used for reactivity feedback and main printouts.

Figure 3.1.2 shows the flow through the pre-voiding core channel thermal hydraulics driver, TSCL0. This routine is entered once for each channel during each coolant time step. The coolant flow rates are calculated before the heat-transfer module (TSHTRN) is called. TSHTRN is only called if the current coolant time step completes a heat-transfer time step. The voiding model, TSHTRV, is described in Chapter 12.

In this chapter, Section 3.2 describes the mesh structure used for heat-transfer calculations. Then, Section 3.3 describes the pre-boiling transient heat-transfer calculations, followed by the steady-state thermal hydraulics calculations in Section 3.4. The pre-voiding transient heat transfer is discussed before the steady-state thermal hydraulics for two reasons. First, the code is primarily a transient code, so the transient calculations are more important. Second, the finite difference approximations were made with the transient calculations in mind, and the steady-state solution was formulated to be consistent with the approximations used in the transient. Section 3.5 describes TSHTRV, the fuel-pin heat-transfer calculations in the boiling module. Section 3.6 describes the treatment of the bond-gap conductance between the fuel and the cladding. Section 3.7 describes modifications to the fuel pin heat transfer calculations for PLUTO2 and LEVITATE. The heat transfer time step control is described in Section 3.8. Section 3.9 describes steady-state and pre-voiding transient hydraulics. Section 3.10 describes the multiple pin option. Subassembly-to-subassembly heat transfers described in Section 3.11. Section 3.12 describes interaction with other modules. It is followed by sections providing subroutine descriptions and flowcharts, subchannel model treatment details, thermal properties of fuel and cladding, and a description of the input to, and output from, the thermal hydraulic routines.

3.2 SAS4A/SASSYS-1 Channel Approach

SAS4A/SASSYS-1 is capable of using a multi-channel treatment. Each channel represents a fuel pin, its associated coolant, and a fraction of the subassembly duct wall, as indicated in Figure 3.2.1. Usually, a channel is used to represent an average pin in a fuel subassembly or a group of similar subassemblies. A channel can also be used to represent a blanket assembly or a control-rod channel, and the hottest pin in a subassembly can be represented instead of the average pin. Different channels can be used to account for radial and azimuthal power variations within the core, as well as variations in coolant flow orificing and fuel burn-up. In the multiple pin option, more than one channel can be used to represent a subassembly.

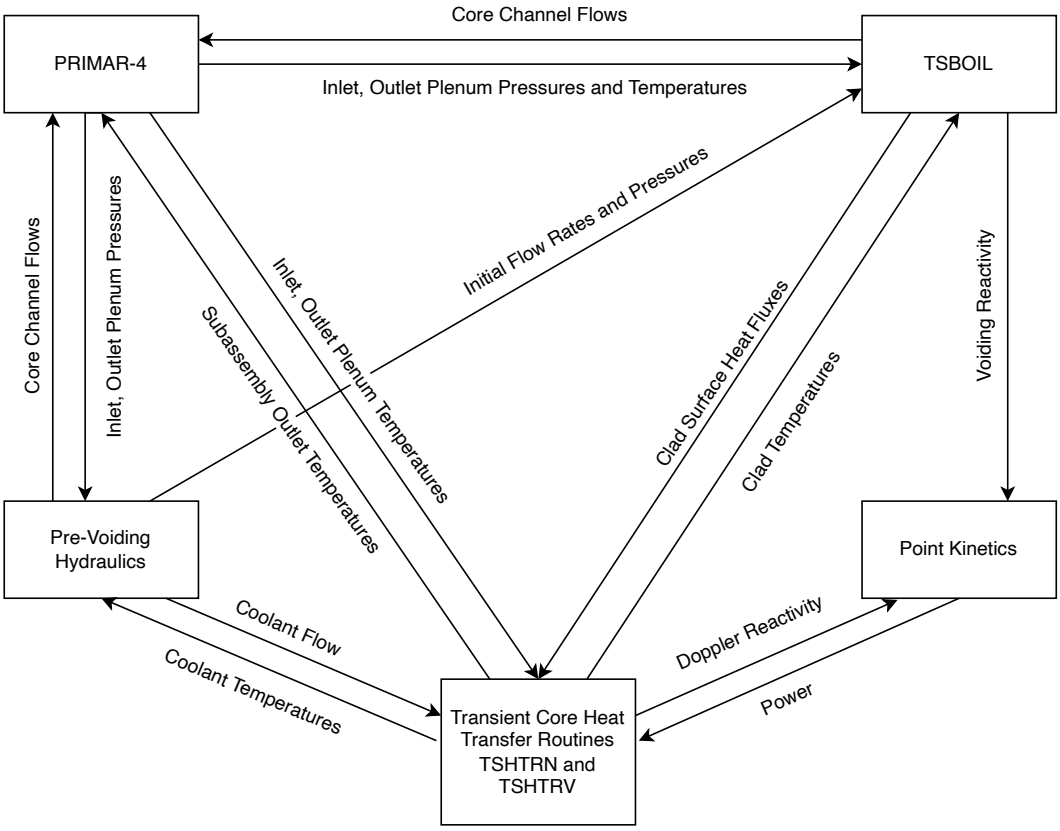


Figure 3.1.1: Interactions of Thermal-Hydraulic Routines with Other Modules

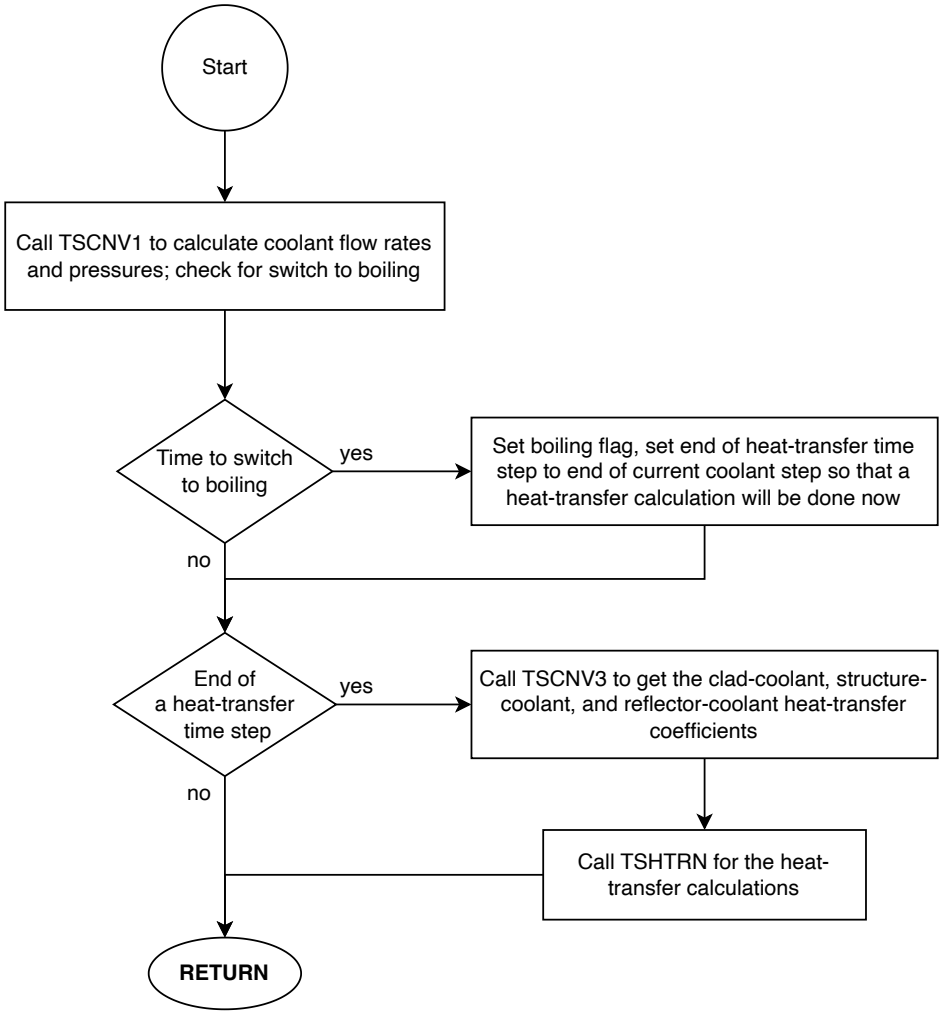


Figure 3.1.2: Flowchart for the Pre-Voiding Core Channel Thermal Hydraulics Driver (Subroutine TSCL0)

3.2.1 Axial Mesh Structure

A channel usually represents the whole length of the subassembly, from coolant inlet to coolant outlet. A number of axial zones are used, as indicated in Figure 3.2.2. One zone represents the fuel-pin section, including the core, axial blankets, and gas plenum. Other zones represent reflector regions above and below the pin section. A maximum of 7 zones can be used in a channel. In general, radial dimensions and thermal properties are constant with a reflector zone. The pin section zone is treated separately in considerably more detail than the reflector zones. The gas plenum can be either above or below the core.

Figure 3.2.3 shows the axial mesh structure used for a channel. The coolant and structure nodes run the whole length of the channel. The coolant nodes are staggered with respect to the fuel, cladding, reflector, and structure nodes. Using coolant temperatures defined at the axial boundaries between cladding and structure nodes makes it easier to calculate accurate coolant temperatures. If non-uniform axial mesh sizes are used, a simple finite differencing of the coolant temperature equation gives accurate coolant temperatures with a staggered mesh, whereas if the coolant nodes were at the middle of the cladding nodes, then obtaining accurate coolant temperatures would require extra terms in the finite differencing of the coolant temperature equation.

3.2.2 Radial Mesh Structure

3.2.2.1 Core and Blanket Region

Figure 3.2.4 shows the radial mesh structure used for temperature calculations in the core and blanket regions. This figure represents one axial node. Between four and eleven radial nodes are used in the fuel, three in the cladding, one in the coolant, and two in the structure. In the fuel, the nodes can be set up on either an equal radial difference basis or an equal mass basis. In either case, the first and last nodes are half-size. For a given number of nodes, an equal radial difference mesh will usually give more accurate center-line temperatures, but equal mass nodes are sometimes used to get more nodes in the outer part of the fuel, where temperature gradients are steeper. Steady-state fuel restructuring can change the node sizes. Also, during the transient calculation, the radii will move with the fuel as it expands or contracts due to temperature changes. After the steady-state initialization, the mass of fuel associated with a radial node is constant, at least until fuel-pin disruption and coupling is made to PLUTO2, PINACLE, or LEVITATE.

The inner fuel node is at $r = 0$ if there is no central void. Otherwise, it is at the fuel inner surface. The outer fuel node is at the fuel outer surface.

The “structure” represents each pin’s share of the duct wall. A wrapper wire can be lumped in with either the cladding or the structure.

3.2.2.2 Gas Plenum Region

The radial mesh structure used in the gas plenum region is shown in Figure 3.2.5. The plenum gas is represented by a single axial and radial node. This gas is in contact with a number of axial cladding nodes. At each axial node, there is one radial node in the cladding, one in the coolant, and two in the structure.

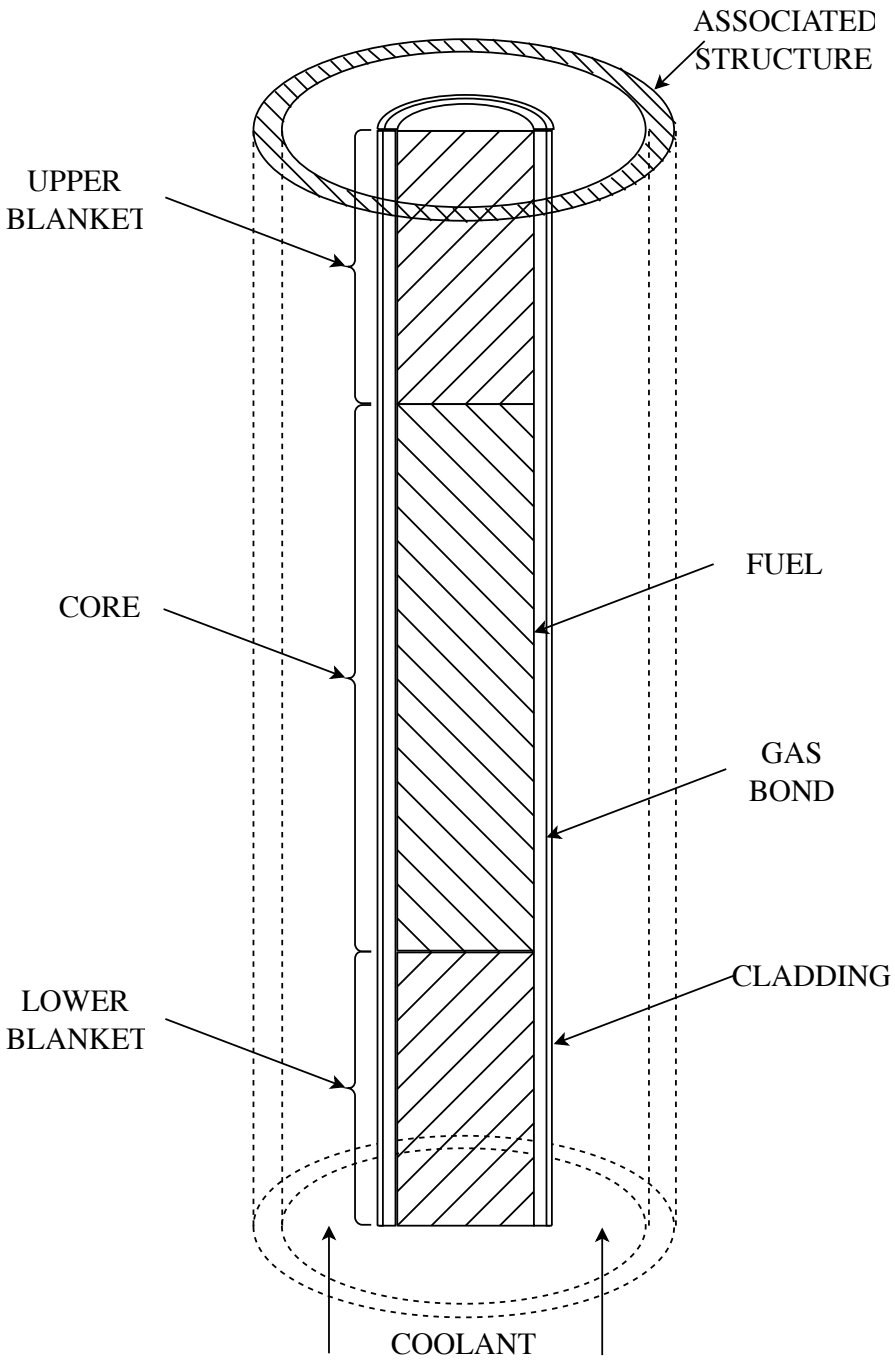


Figure 3.2.1: SAS4A/SASSYS-1 Channel Treatment

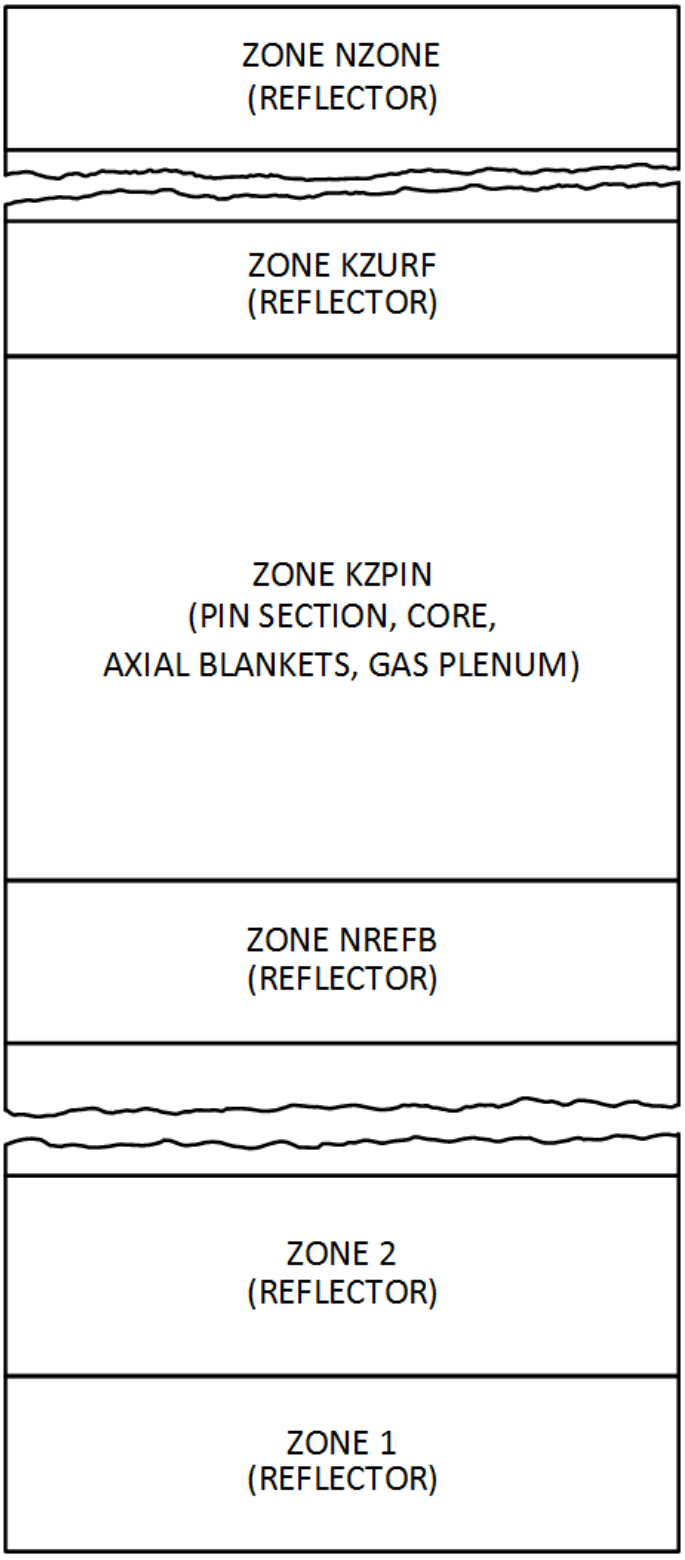


Figure 3.2.2: Axial Zones in a SAS4A/SASSYS-1 Channel

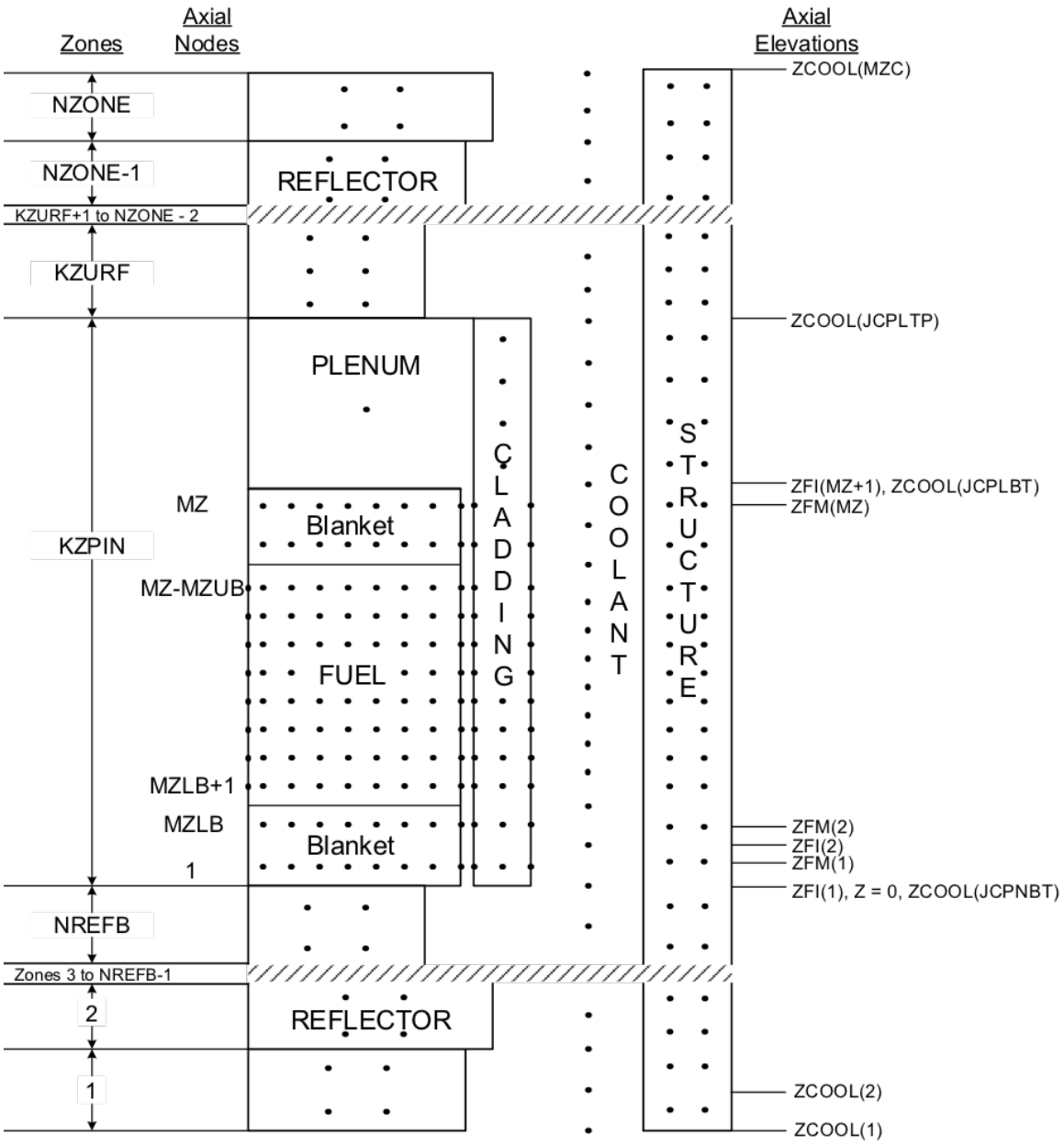


Figure 3.2.3: Schematic of SAS4A/SASSYS-1 Channel Discretization

3.2.2.3 Reflector Regions

The radial mesh structure in an axial node in a reflector region is shown in Figure 3.2.6. Two nodes are used in the reflector, one in the coolant, and two in the structure.

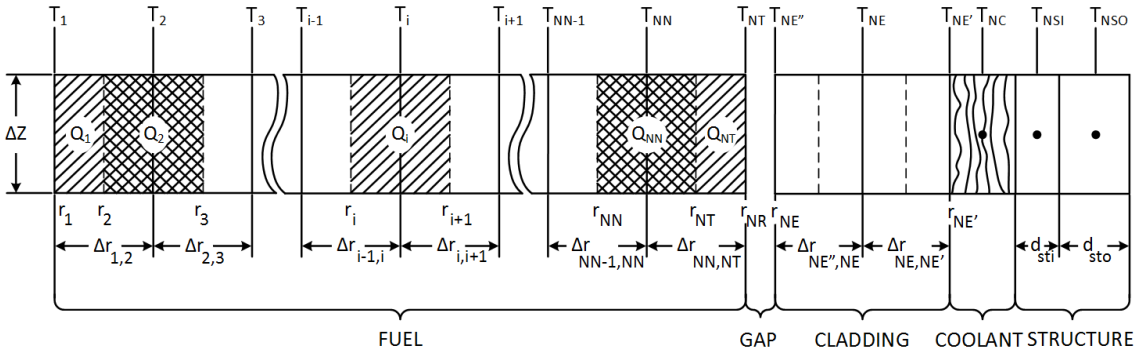


Figure 3.2.4: Radial Temperature Nodes, Core and Axial Blanket Regions

3.3 Pre-boiling Transient Heat Transfer, Single Pin Model

The transient fuel-pin temperature calculations in SAS4A/SASSYS-1 are similar to the Crank-Nicolson scheme [3-1] used in SAS2A [3-2] and SAS3D [3-3], but there are a number of significant differences. One reason for these differences is to allow the use of larger heat-transfer time steps with less computing time per step. Another reason is to obtain greater accuracy and more precise energy balance.

In order to use very long heat-transfer time steps on one second or more in the pre-voiding phase of a very slow transient, the fuel, cladding, coolant, and structure temperatures at an axial node are computed simultaneously in non-voiding cases. Therefore, coolant temperatures are computed in the fuel-pin heat-transfer routine in the non-voiding situation.

The coupling between TSHTRN and the pre-voiding coolant dynamics routines is different from the coupling between TSHTRV and the voiding routines, although in both cases the coolant dynamics or voiding calculations are done before the fuel-pin heat-transfer calculations. In the non-voiding case, extrapolated coolant temperatures are used to obtain the temperature-dependent sodium properties used in the calculation of the coolant flow rates and heat-transfer coefficients. Then, TSHTRN uses these values to compute the coolant temperatures, and these new coolant temperatures are used in the next extrapolation. In the voiding routines, extrapolated cladding temperatures are used in the implicit calculation of the heat flux from cladding to coolant to obtain new coolant temperatures. The voiding routines then sum the integrated heat flux from cladding to coolant at each axial node, and this integrated heat flux at the cladding surface is used as a boundary condition in TSHTRV.

TSHTRN and TSHTRV are somewhat simpler than the corresponding TSHTR in SAS3A and SAS3D. In SAS3D, TSHTR contains extraneous material related to other modules, such as initiating cladding and fuel motion; this material is not included in the SAS4A/SASSYS-1 heat-transfer routines. Also, in SAS3D, the fuel mass for each radial node of each axial node is recomputed every time step from the temperature-dependent fuel density and the node radii. In SAS4A/SASSYS-1 the node mass is computed in the steady-state module and then stored for used in the transient. This node mass is held constant until fuel relocation starts.

Another difference between the SAS3D and SASSYS-1 heat-transfer routines is in the amount of vectorization of the algorithms and the coding. Some of the calculations in the SAS3D routines happen to vectorize, but a special effort was made to vectorize many more of the calculations in the SAS4A/SASSYS-1 heat-transfer routines. In general, vectorizing involves setting up arrays so that all elements in the array are processed in the

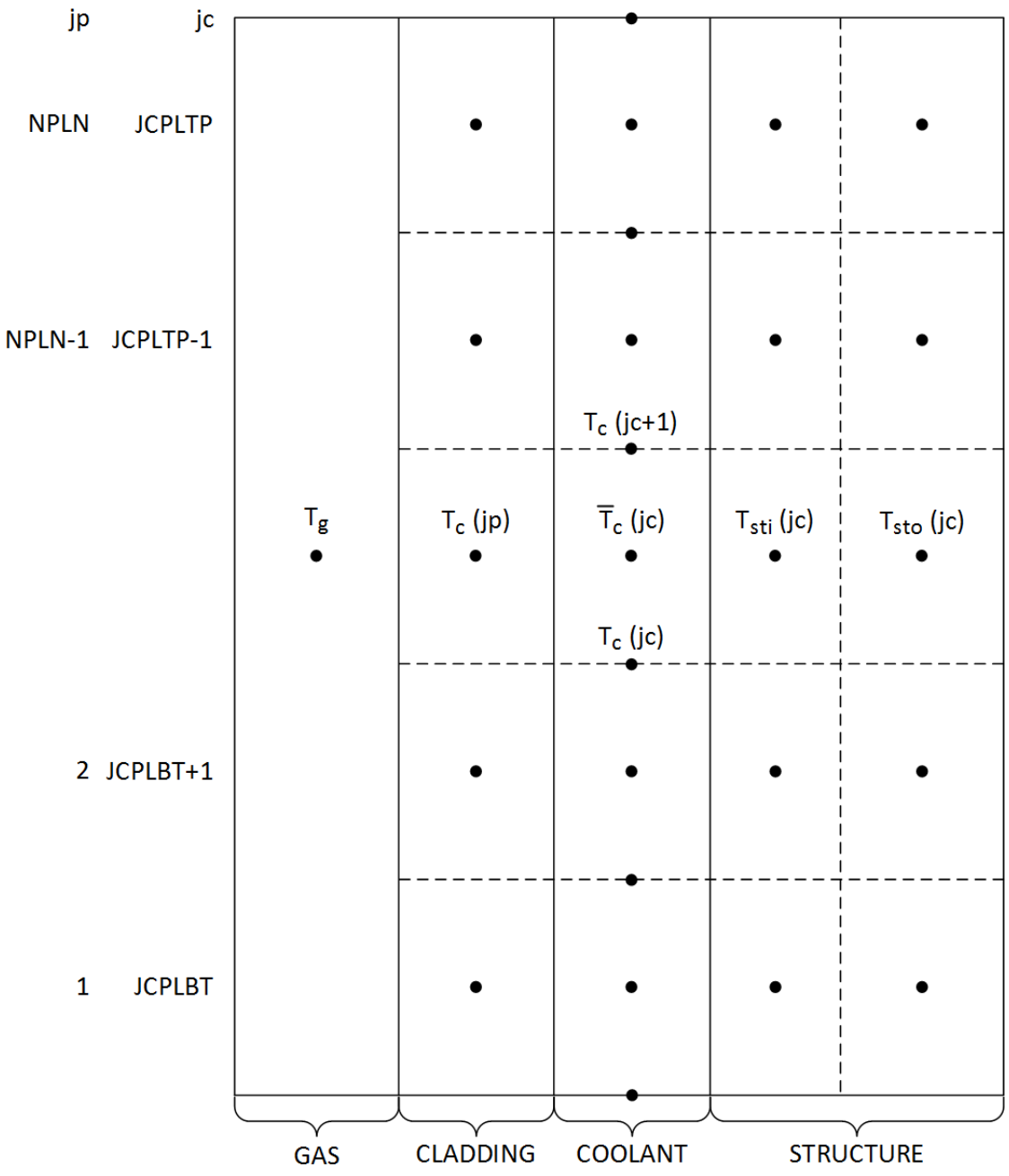


Figure 3.2.5: Radial Temperature Nodes, Gas Plenum Region

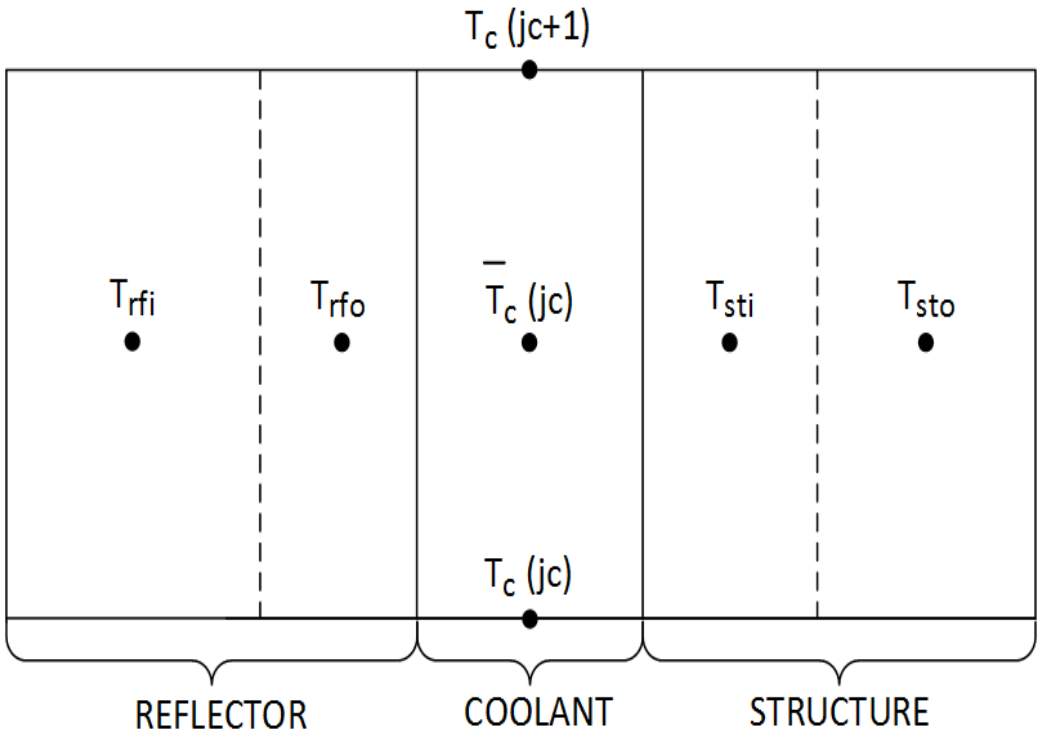


Figure 3.2.6: Radial Temperature Nodes, Reflector Region

same way and so that the results of a given calculation for one element in the array do not depend on the results for any other element in the array. Vectorization is partly just a coding matter, but it also involves choosing vectorizable algorithms or adapting algorithms for vectorization.

To some extent, the descriptions in the following sections reflect the emphasis on vectorization. Intermediate quantities used in the solution are usually defined as elements in arrays, and, where possible, the array elements are defined so that all of the elements in an array can be treated in the same manner in the calculations. In order to emphasize the array nature of the solution, when a new array is introduced, all of the elements in the array are defined in the same section, even though many of the elements are often not used until later sections.

As indicated above in Section 3.2.2, three different radial mesh structures are used in the heat-transfer calculations: one for the core and blanket region, one for the gas plenum region, and one for the reflector regions. Different heat-transfer calculations are done for each of these three regions.

3.3.1 Core and Axial Blankets

3.3.1.1 Basic Equations

The basic transient heat-transfer equation within the fuel and within the cladding is

$$\rho c \frac{\partial \mathrm{T}}{\partial \mathrm{t}} = \frac{1}{r} \frac{\partial}{\partial \mathrm{r}} \left(\mathrm{kr} \frac{\partial \mathrm{T}}{\partial \mathrm{r}} \right) + Q \tag{3.3-1}$$

where

T = temperature

ρ = density

c = specific heat

r = radius

k = thermal conductivity

t = time

and

Q = heat source per unit volume

Melting of fuel and cladding is treated using a melting range, bounded by solidus and liquidus temperatures, T_{sol} and T_{liq}, respectively, and a heat of fusion, U_{melt}. Separate values of T_{sol}, T_{liq}, and U_{melt} are used for the fuel and cladding. In the melting range, Eq. (3.3-1) is modified and becomes

$$\rho c_{\mathrm{m}} \frac{\partial \mathrm{T}}{\partial \mathrm{t}} = \frac{1}{r} \frac{\partial}{\partial \mathrm{r}} \left(\mathrm{kr} \frac{\partial \mathrm{T}}{\partial \mathrm{r}} \right) + Q \tag{3.3-2}$$

where

$$c_{\mathrm{m}} = \frac{U_{\mathrm{melt}}}{T_{\mathrm{liq}} - T_{\mathrm{sol}}} \tag{3.3-3}$$

In the actual solution of Eq. (3.3-2) for a time step, c is used instead of c_{m} in the calculation of the temperature change for the step. Then, if the temperatures are in the melting range, the computed temperature change is modified to account for the heat of fusion, as described in Section 3.3.5.

The heat flux, q_{fe}, from the fuel outer surface to the cladding inner surface contains both a bond gap conductance term, h_{b}, and a radiation term:

$$q_{\mathrm{fe}} = h_{\mathrm{b}} \left[T\left(\mathrm{NT}\right) - T\left(\mathrm{NE}''\right) \right] + \epsilon\sigma \left[T\left(\mathrm{NT}\right)^4 - T\left(\mathrm{NE}''\right)^4 \right] \tag{3.3-4}$$

where

ϵ = thermal emissivity of the fuel

σ = Stefan-Boltzmann constant

T(NT) = fuel outer surface temperature

and

T(NE") = cladding inner surface temperature

For the coolant, the basic heat-transfer equation is

$$\rho c\, A_{\mathrm{c}} \frac{\partial \mathrm{T}}{\partial \mathrm{t}} + \frac{\partial}{\partial \mathrm{z}} \left(\mathrm{wcT}\right) = \left(Q_{\mathrm{c}} + Q_{\mathrm{ec}} + Q_{\mathrm{sc}}\right) A_{\mathrm{c}} \tag{3.3-5}$$

where A_{c} = coolant flow area. The heat, Q_{c}, produced directly in the coolant is computed from the fraction, γ_{c}, of the total energy production that goes into neutron and gamma heating of the coolant:

$$Q_{\mathrm{c}} = \frac{\gamma_{\mathrm{c}} \overline{P}(j)}{A_{\mathrm{c}} \Delta z(j)} \tag{3.3-6}$$

where

$\overline{P}\,(j)$ = total heat production rate in axial node j

and

$\Delta z(j)$ = axial node height

The heat flow from the cladding to the coolant, Q_{ec} is calculated as

$$Q_{\mathrm{ec}} = h_{\mathrm{c}} \left[T\left(\mathrm{NE}'\right) - T(\mathrm{NC})\right] \frac{2\pi r(\mathrm{NE}')}{A_{\mathrm{c}}} \tag{3.3-7}$$

and the heat flow from structure to coolant, Q_{sc}, is calculated as

$$Q_{\mathrm{sc}} = h_{\mathrm{c}} \left[T\left(\mathrm{NSI}\right) - T(\mathrm{NC})\right] \frac{S_{\mathrm{st}}}{A_{\mathrm{c}}} \tag{3.3-8}$$

where S_{st} is the perimeter of the structure. The coolant heat-transfer coefficient, h_c, is calculated using

$$\frac{h_{\mathrm{c}} D_{\mathrm{h}}}{k} = c_1 \left[\frac{D_{\mathrm{h}}\ \mathrm{w}\ \mathrm{c}}{\mathrm{k}\ A_{\mathrm{c}}}\right]^{c_2} + \ c_3 \tag{3.3-9}$$

which is a form used in correlations for convective heat-transfer coefficients for low Prandtl number fluids, such as liquid metal [3-4]. The user supplied constants c_1, c_2, and c_3 depend on the particular correlation used.

The structure is treated with a one-dimensional heat conduction equation:

$$\rho c \frac{\partial \mathrm{T}}{\partial \mathrm{t}} = \frac{\partial}{\partial \mathrm{x}} \left[k\, \frac{\partial \mathrm{T}}{\partial \mathrm{x}}\right] + Q \tag{3.3-10}$$

The treatment of the heat source, Q, in the structure is discussed in Section 3.3.1.2.8.

3.3.1.2 Finite Difference Equations

Finite differencing in both space and time is used for the transient heat-transfer calculations. The radial mesh structure is described in Section 3.2.2 above. In the equations below, the time t represents the beginning of the temp step, and Δt is the step size. The parameters θ_1 and θ_2 determine the degree of implications. For an explicit scheme, θ_1 = 1.0 and θ_2 = 0.0. For a fully implicit scheme, θ_1 = 0.0 and θ_2 = 1.0. For a semi-implicit scheme θ_1 = 0.5 and θ_2 = 0.5. The degree of implicitness is calculated by the code, based on the ratio of the time-step size, Δt, to a user-supplied time constant for fuel-pin heat transfer, τ_{ht}. As explained in Section 3.19.1, the expression used for θ_1 and θ_2 in TSHTRN are

$$\theta_2 = \frac{1.65 + x}{3.3 + x} \tag{3.3-11}$$

where

$$x = \frac{\Delta t}{\tau_{\text{ht}}} \tag{3.3-12}$$

and

$$\theta_1 = 1.0 - \theta_2 \tag{3.3-13}$$

Also, large relative changes in coolant flow rate, w, during a long heat transfer time step can lead to anomalous coolant temperature changes if the value of θ_1 is too large; therefore, θ_2 is calculated as

$$\theta_2 = \frac{|w_1|}{|w_1| + |w_2|} \tag{3.3-14}$$

if Eq. (3.3-14) gives a larger value than Eq. (3.3-11). In Eq. (3.3-14), w_1 and w_2 are the coolant flow rates at the beginning and end of the heat-transfer time step, respectively. In any case, Eq. (3.3-13) is used for θ_1.

The heat transfer time step size is limited to a relatively small value (0.02 s or less) after the onset of boiling, and so TSHTRV used θ_1 = θ_2 = 0.5.

The finite difference equations used for the fuel and for the two inner cladding nodes are the same in TSHTRN and TSHTRV. The differences between these routines start at the outer cladding node.

In general, the time derivative of a variable y is approximated as

$$\frac{\partial \text{y}}{\partial \text{t}} = \frac{y\,(t + \Delta t) - y\,(t)}{\Delta t} \tag{3.3-15}$$

and the spatial derivative is approximated as

$$\frac{\partial \text{y}}{\partial \text{z}} \simeq \theta_1 \left[\frac{y\,(t, z + \Delta z) - y\,(t, z)}{\Delta z}\right] + \theta_2 \left[\frac{y\,(t + \Delta t, z + \Delta z) - y\,(t + \Delta t, z)}{\Delta z}\right] \tag{3.3-16}$$

In the following sections, it will be useful to refer to Figure 3.2.4 for the radial node structure and the definitions of the radial node indexes.

3.3.1.2.1 Fuel Inner Surface, Node 1

There is an adiabatic boundary at the fuel inner surface; so the node 1, Eq. (3.3-1) becomes

$$\begin{aligned} m_{\text{f}}\,(1)\ \overline{c}_{\text{f}}\,(1) \left[\frac{T_2\,(1) - T_1\,(1)}{t}\right] \\ = \frac{2\pi r\,(2)\,\Delta z\,(j)\,\overline{k}_{1,2}}{\Delta r_{1,2}} \{\theta_1\,[T_1\,(2) - T_1\,(1)] + \\ \theta_2\,[T_2\,(2) - T_2\,(1)]\} + Q(1) \end{aligned} \tag{3.3-17}$$

where

$m_f(i)$ = fuel mass at node *i*

$\overline{c}\,(i)$ = fuel heat capacity at radial node i at $t = t_1 + \theta_2 \Delta t$

$T_2(i)$ = temperature at $t + \Delta t$

$T_1(i)$ = temperature at time t

$\Delta z(j)$ = axial mesh height

$$\Delta r(1) = 2\,[r\,(2) - r\,(1)] \tag{3.3-18}$$

$$\Delta r\,(i) = r\,(i+1) - r\,(i) \qquad 2 \;\leq\; i \;\leq\; \text{NN} \tag{3.3-19}$$

$$\Delta r\,(\text{NT}) = 2\,[r\,(\text{NR}) - r\,(\text{NT})] \tag{3.3-20}$$

$$\Delta r\,(i) = \frac{[r\,(\text{NE}') - r\,(\text{NE})]}{2} \quad i = \text{NE}'',\ \text{NE},\ \text{NE}' \tag{3.3-21}$$

(note: NE$''$ = NR)

$$\Delta r_{\text{i,i+1}} = \frac{\Delta r\,(i) + \Delta r\,(i+1)}{2} \;\; 1 \;\leq\; i \;\leq\; \text{NE} \tag{3.3-22}$$

$$Q\,(i) = \frac{\overline{P}\,(j)\,(1 - \gamma_{\text{e}} - \gamma_{\text{c}} - \gamma_{\text{s}})\;\; P_{\text{r}}\,(i)\, m_{\text{f}}\,(i)}{\sum_{\text{ii}=1}^{\text{NT}}\; P_{\text{r}}\,(\text{ii})\, m_{\text{f}}\,(\text{ii})} \;\; 1 \;\leq\; i \;\leq\; \text{NT} \tag{3.3-23}$$

$$Q\,(\text{NE}'') = \frac{\overline{P}\,(j)\,\gamma_{\text{e}}}{4} \tag{3.3-24}$$

$$Q\,(\text{NE}) = \frac{\overline{P}\,(j)\,\gamma_{\text{e}}}{2} \tag{3.3-25}$$

$$Q\,(\text{NE}') = \frac{\overline{P}\,(j)\,\gamma_{\text{e}}}{4} = Q\,(\text{NE}'') \tag{3.3-26}$$

where

$\overline{P}\,(j)$ = total power (watts) in axial node *j*

$P_{\text{r}}(i)$ = radial power shape per unit mass

and

$\gamma_{\text{e}}, \gamma_{\text{c}}, \gamma_{\text{s}}$ = fraction of power in direct heating of clad, coolant, and structure, respectively.

The thermal conductivity, $\overline{k}_{1,2}$ used in Eq. (3.3-17) is a weighted average of the values for the two adjacent nodes. It is calculated as

$$\overline{k}_{\text{i,i+1}} = \frac{k\,(i)\,k\,(i+1)\;\;[\Delta r\,(i) + \Delta r\,(i+1)]}{k\,(i)\,\Delta r\,(i+1) + k\,(i+1)\,\Delta r\,(i)} \tag{3.3-27}$$

where

$k(i)$ = thermal conductivity for radial node *i*, evaluated using the fuel temperature extrapolated to $t + \theta_2 \Delta t$

Eq. (3.3-27) is carried over from SAS3D. The fuel restructuring algorithm used in SAS3D uses up to three different fuel types (columnar, equiaxed, and unrestructured) for the fuel at each axial node. Sharp boundaries between fuel types are used. The radial mesh is adjusted, if necessary, so that fuel-type boundaries fall on radial node boundaries. One consequence of the sharp fuel-type boundaries in SAS3D is that the fuel thermal conductivity can change significantly from one node to the next at a fuel-type boundary. The average thermal

conductivity of Eq. (3.3-27) will give accurate steady-state fuel temperatures even if the thermal conductivity has large jumps at node boundaries. The fuel restructuring provided by DEFORM-IV in SAS4A is somewhat smoother than that used in SAS3D, and the node-to-node changes in thermal conductivity in SAS4A are usually smaller than the corresponding changes at fuel-type boundaries in SAS3D, so the need for special weighting of the fuel thermal conductivities in SAS4A Lis less than in SAS3D, but Eq. (3.3-27) still provides more accurate fuel temperatures than simpler weighting schemes.

Note that even though most of the variables in Eq. (3.3-17) to Eq. (3.3-27) vary with axial node j, the subscript j has only been included for some of these variables.

3.3.1.2.2 Inner Fuel Nodes, Nodes 2 to NN

For fuel radial node I, Eq. (3.3-1) becomes

$$\begin{aligned} m_{\mathrm{f}}(i)\,\overline{c}_{\mathrm{f}}(i)\;\frac{[T_2(i)-T_1(i)]}{\Delta t} \\ = \frac{2\pi r(i+1)\,\Delta z(j)\,\overline{k}_{\mathrm{i,i+1}}}{\Delta r_{\mathrm{i,i+1}}}\left\{\theta_1\;[T_1(i+1)-T_1(i)]+\theta_2\;[T_2(i+1)-T_2(i)]\right\} \\ -\frac{2\pi r(i)\,\Delta z(j)\,\overline{k}_{\mathrm{i-1,i}}}{\Delta r_{\mathrm{i-1,i}}}\left\{\theta_1\;[T_1(i)-T_1(i-1)]+\theta_2\;[T_2(i)-T_2(i-1)]\right\}+Q(i) \end{aligned} \tag{3.3-28}$$

Note that the left-hand side of Eq. (3.3-28) represents the change in internal energy as node i, whereas the terms on the right-hand side represent heat conduction into node i from nodes $i+1$ and $i-1$, as well as the heat source in node i.

3.3.1.2.3 Fuel Outer Surface Node, Node NT

The heat flux q_{fc}, from the fuel outer surface to the cladding inner surface contains both a bond gap conductance and a radiation term.

$$q_{\mathrm{fc}} = h_{\mathrm{b}}\left[\mathrm{T(NT)}-\mathrm{T(NE'')}\right]+\varepsilon\sigma\;\left[\mathrm{T(NT)}^4-\mathrm{T(NE'')}^4\right] \tag{3.3-29}$$

where

h_{b} = bond conductance

ϵ = thermal emissivity of the fuel

and

σ = Stefan-Boltzmann constant.

The T^4 terms are rewritten as

$$T(\mathrm{NT})^4 - T(\mathrm{NE''})^4 = h_{\mathrm{r}}\left[T(\mathrm{NT})-T(\mathrm{NE''})\right] \tag{3.3-30}$$

where

$$h_{\mathrm{r}} = \frac{T_1(\mathrm{NT})^4 - T_1(\mathrm{NE''})^4}{T_1(\mathrm{NT})-T_1(\mathrm{NE''})} = \left[T_1(\mathrm{NT})+T_1(\mathrm{NE''})\right]\left[T_1(\mathrm{NT})^2+T_1(\mathrm{NE''})^2\right] \tag{3.3-31}$$

The approximation is then made that h_{r} is a constant for a time step, and the equation for node NT becomes

$$\begin{aligned} m_{\mathrm{f}}(\mathrm{NT})\,\overline{c}_{\mathrm{f}}(\mathrm{NT})\;\frac{T_2(\mathrm{NT})-T_1(\mathrm{NT})}{\Delta t} = \frac{2\pi r(\mathrm{NT})\,\Delta z(j)\,\overline{k}_{\mathrm{NN,NT}}}{\Delta r_{\mathrm{NN,NT}}}\left\{\theta_1\left[T_1(\mathrm{NN})-T_1(\mathrm{NT})\right]+\right. \\ \left.\theta_2\left[T_2(\mathrm{NN})-T_2(\mathrm{NT})\right]\right\}\;+2\pi r(\mathrm{NR})\,\Delta z(j)\left[h_{\mathrm{b}}+\epsilon\sigma h_{\mathrm{r}}\right]\;\left\{\theta_1\left[T_1(\mathrm{NE''})-T_1(\mathrm{NT})\right]\right. \\ \left.+\theta_2\left[T_2(\mathrm{NE''})-T_2(\mathrm{NT})\right]\right\}+Q(\mathrm{NT}) \end{aligned} \tag{3.3-32}$$

3.3.1.2.4 Cladding Inner Node

For the cladding inner node, NE$''$, Eq. (3.3-1) becomes

$$\begin{aligned}
&\frac{m_{\mathrm{e}}c_{\mathrm{e}}}{4}\frac{T_2\left(\mathrm{NE}''\right)-T_1\left(\mathrm{NE}''\right)}{\Delta t} \\
&=\frac{2\pi\Delta z\left(j\right)\overline{r}_{\mathrm{NE}}\,k_{\mathrm{NE}'',\mathrm{NE}}}{\Delta r_{\mathrm{NE}'',\mathrm{NE}}}\left\{\theta_1\ \left[T_1\left(\mathrm{NE}\right)-T_1\left(\mathrm{NE}''\right)\right]+\theta_2\ \left[T_2\left(\mathrm{NE}\right)-T_2\left(\mathrm{NE}''\right)\right]\right\} \\
&\quad -2\pi r\left(\mathrm{NR}\right)\Delta z\left(j\right)\left(h_{\mathrm{b}}+\varepsilon\sigma h_{\mathrm{r}}\right)\left\{\theta_1\ \left[T_1\left(\mathrm{NE}''\right)-T_1\left(\mathrm{NT}\right)\right]\right. \\
&\quad \left.+\theta_2\ \ \left[T_2\left(\mathrm{NE}''\right)-T_2\left(\mathrm{NT}\right)\right]\right\}+Q\left(\mathrm{NE}''\right)
\end{aligned} \tag{3.3-33}$$

where

c_{e} = cladding heat capacity

$$m_{\mathrm{e}}=\text{cladding mass}=2\pi\rho_{\mathrm{e}}\left[r\left(\mathrm{NE}'\right)^2-r\left(\mathrm{NE}\right)^2\right]\Delta z(j) \tag{3.3-34}$$

p_{e} = cladding density

and

$$\overline{r}_{\mathrm{NE}}=r\ \left(\mathrm{NE}\right)+\frac{1}{4}\left[r\ \left(\mathrm{NE}'\right)-r\ \left(\mathrm{NE}\right)\right] \tag{3.3-35}$$

Note that the factors of 4 in Eq. (3.3-33) and Eq. (3.3-35) come about because the inner cladding node represents one fourth of the thickness of the cladding.

3.3.1.2.5 Cladding Mid-point Node

For the cladding mid-point node, NE, Eq. (3.3-1) becomes

$$\begin{aligned}
&\frac{m_{\mathrm{e}}c_{\mathrm{e}}}{2}\frac{T_2\left(\mathrm{NE}\right)-T_1(\mathrm{NE})}{\Delta t} \\
&=\frac{2\pi\overline{r}_{\mathrm{NE}'}\Delta z(j)\overline{k}_{\mathrm{NE,NE}'}}{\Delta r_{\mathrm{NE,NE}'}}\left\{\theta_1\left[T_1\left(\mathrm{NE}'\right)-T_1(\mathrm{NE})\right]+\theta_2\left[T_2\left(\mathrm{NE}'\right)-T_2(\mathrm{NE})\right]\right\} \\
&-\frac{2\pi\overline{r}_{\mathrm{NE}}\Delta z(j)\overline{k}_{\mathrm{NE}'',\mathrm{NE}}}{\Delta r_{\mathrm{NE}'',\mathrm{NE}}}\left\{\theta_1\left[T_1\left(\mathrm{NE}\right)-T_1(\mathrm{NE}'')\right]+\theta_2\left[T_2\left(\mathrm{NE}\right)-T_2(\mathrm{NE}'')\right]\right\}+Q(\mathrm{NE})
\end{aligned} \tag{3.3-36}$$

where

$$\overline{r}_{\mathrm{NE}'}=r\ \left(\mathrm{NE}\right)+\frac{3}{4}\left[r\left(\mathrm{NE}'\right)-r\ \left(\mathrm{NE}\right)\right] \tag{3.3-37}$$

3.3.1.2.6 Cladding Outer Node

The outer cladding node, NE$'$, transfers heat to both the cladding mid-point node and the coolant node, so the equation for the outer cladding node temperature is

$$\begin{aligned}
&\frac{m_{\mathrm{e}}c_{\mathrm{e}}}{4}\frac{T_2\left(\mathrm{NE}'\right)-T_1(\mathrm{NE}')}{\Delta t} \\
&=-\frac{2\pi\overline{r}_{\mathrm{NE}'}\Delta z(j)\overline{k}_{\mathrm{NE,NE}'}}{\overline{r}_{\mathrm{NE,NE}'}}\ \left\{\theta_1\left[T_1\left(\mathrm{NE}'\right)-T_1(\mathrm{NE})\right]+\theta_2\left[T_2\left(\mathrm{NE}'\right)-T_2(\mathrm{NE})\right]\right\} \\
&+2\pi r\left(\mathrm{NE}'\right)\Delta z(j)\ \left\{\theta_1 h_{\mathrm{c1}}(j)\left[T_1\left(\mathrm{NC}\right)-T_1\left(\mathrm{NE}'\right)\right]+\theta_2 h_{\mathrm{c2}}(j)\left[T_2\left(\mathrm{NC}\right)-T_2\left(\mathrm{NE}'\right)\right]\right\}+Q(\mathrm{NE}')
\end{aligned} \tag{3.3-38}$$

Note that $T(\text{NC}) = \overline{T}_c\,(jc)$. Also, hc_1 and hc_2 are the coolant heat-transfer coefficients at t and $t + \Delta t$ as calculated from Eq. (3.3-9),

3.3.1.2.7 Coolant, Node NC

Coolant temperatures are calculated for the whole length of the subassembly, whereas fuel temperatures are computed in the core and blankets only. Therefore, the axial coolant node mesh extends beyond the fuel mesh; and the coolant axial node index, jc, is related to the fuel axial node index, j, by

$$jc = j + j_{\text{cblbt}} - 1 \tag{3.3-39}$$

where j_{cblbt} is the coolant node at the bottom of the lower blanket. In the $T_1(i,j)$ and $T_2(i,j)$ arrays, the coolant node corresponds to

$$T_2\,(\text{NC,j}) = \overline{T}_{\text{c}}\,(jc) \tag{3.3-40}$$

In the non-voiding case, the coolant flow is usually upward. In such a situation, the transient calculation for a time step starts at the subassembly inlet and works upward through the lower reflector zones, through the pin section, and finally through the upper reflector zones. The code can also handle downward coolant flow during the transient, although the initial steady-state coolant flows must all be upward. In the downward situation, the calculation starts at the top of the subassembly and works down to the inlet.

In the core and blanket regions, Eq. (3.3-5) becomes

$$\begin{aligned}\overline{\rho}_{\text{c}}\,(jc)\,\overline{c}_{\text{c}}\,(jc)\,\frac{T_2\,(\text{NC}) - T_1\,(\text{NC})}{\Delta t} + \frac{2\overline{c}_{\text{c}}\,(jc)}{\Delta z\,(j)\,A_{\text{c}}\,(jc)}\;\{\theta_1\,|w_1|\,[T_1\,(\text{NC}) - T'_1] \\ +\theta_2\,|w_2|\,[T_2\;(\text{NC}) - T'_2]\,\} = \frac{Q\,(\text{NC})}{A_{\text{c}}\,(jc)\,\Delta z\,(j)} + \frac{2\pi r\,(NE')}{A_{\text{c}}\,(jc)} \\ \{\theta_1\,h_{\text{c1}}\,(j)\,[T_1\,(\text{NE}') - T_1\,(\text{NC})] + \theta_2 h_{\text{c2}}\,(j)\,[T_2\;(NE') - T_2\,(\text{NE})]\} \\ +\frac{S_{\text{pr}}}{A_{\text{c}}\,(jc)}\,\{\theta_1\,H_{\text{sic1}}\,(jc)\,[T_1\,(\text{NSI}) - T_1\,(\text{NC})] \\ +\theta_2\,H_{\text{sic2}}\,(jc)\,[T_2\,(\text{NSI}) - T_2\,(\text{NC})]\}\end{aligned} \tag{3.3-41}$$

where

S_{pr} = structure perimeter

w_1 and w_2 = the coolant mass flow rates (kg/s) at t and $t + t$

NSI = the inner structure node

$$T'_1 = \begin{cases} T_{\text{c1}}\,(jc) & \text{if } w_1 \geq 0 \\ T_{\text{c}}\,(jc+1) & \text{if } w_1 < 0 \end{cases} \tag{3.3-42}$$

$$T'_2 = \begin{cases} T_{\text{c2}}\,(jc) & \text{if } w_2 \geq 0 \\ T_{\text{c}}\,(jc+1) & \text{if } w_2 < 0 \end{cases} \tag{3.3-43}$$

and

H_{sic} = the heat-transfer coefficient from the structure inner node to the coolant

$$H_{\text{sic}} = \frac{2h_{\text{c}}k_{\text{si}}}{2k_{\text{si}} + h_{\text{c}}d_{\text{sti}}} \tag{3.3-44}$$

Note that Eq. (3.3-44) is obtained by adding thermal resistance in series:

$$\frac{1}{H_{\text{sic}}} = \frac{1}{h_{\text{c}}} + \frac{d_{\text{sti}}}{2k_{\text{si}}} \tag{3.3-45}$$

3.3.1.2.8 Structure Inner Node, Node NSI

In the core and blanket regions,

$$\begin{aligned}(\rho c)_{\text{sti}}\, d_{\text{sti}} \frac{T_2\,(\text{NSI}) - T_1\,(\text{NSI})}{\Delta t} &= \theta_1 H_{\text{sic1}}\,(jc)\,[T_1\,(\text{NC}) - T_1\,(\text{NSI})] \\ &+\theta_2\, H_{sic2}\,(jc)\,[T_2\ \,(\text{NC}) - T_2\,(\text{NSI})] \\ &+H_{\text{stio}}\,(jc)\,\{\theta_1\,[T_1\,(\text{NSO}) - T_1\,(\text{NSI})] \\ &+\theta_2\,[T_2\,(\text{NSO}) - T_2\,(\text{NSI})]\} + Q_{\text{st}} \frac{d_{\text{sti}}}{d_{\text{sti}} + d_{\text{sto}}}\end{aligned} \tag{3.3-46}$$

where

NSO = outer structure node

d_{sti} = thickness of inner structure node

d_{sto} = thickness of outer structure node

$$H_{\text{stio}} = \frac{2k_{\text{si}}k_{\text{so}}}{d_{\text{sti}}k_{\text{so}} + d_{\text{sto}}k_{\text{si}}} \tag{3.3-47}$$

k_{si} = thermal conductivity of the inner structure node

k_{so} = thermal conductivity of the outer structure node

and

$Q_{\text{st}} = \frac{\gamma_s \overline{P}(j)}{S_{\text{pr}}\Delta z(j)}$ = direct heating source in the structure.

The left-hand side of Eq. (3.3-46) represents the change in internal energy in the node. The terms on the right-hand side represent heat flow from the coolant and the outer structure node, as well as direct heating of the structure by neutrons and gamma rays. It is assumed that the direct heating source is divided between the inner and outer nodes in proportion to their thicknesses.

3.3.1.2.9 Structure Outer Node, Node NSO

In the core and blankets,

$$\begin{aligned}(\rho c)_{\text{sto}}\, d_{\text{sto}} \frac{T_2\,(\text{NSO}) - T_1(\text{NSO})}{\Delta t} &= H_{\text{stio}}\,(jc)\,\{\theta_1\,[T_1\,(\text{NSI}) - T_1\,(\text{NSO})] \\ &+\theta_2\,[T_2\,(\text{NSI}) - T_2(\text{NSO})]\} + Q_{\text{st}}\,(jc)\,\frac{d_{\text{sto}}}{d_{\text{sti}} + d_{\text{sto}}} + \frac{Q_{\text{chch}}(jc)}{S_{\text{pr}}}\end{aligned} \tag{3.3-48}$$

where

$Q_{\text{chch}} = Q_{\text{chch,I}}$ = subassembly-to-subassembly linear heat transfer rate.

The values used for Q_{chch} are discussed in Section 3.11.

3.3.1.2.10 Solution of Finite Difference Equations

Eq. (3.3-17), Eq. (3.3-28), Eq. (3.3-32), Eq. (3.3-33), Eq. (3.3-36), Eq. (3.3-38), Eq. (3.3-41), Eq. (3.3-46), and Eq. (3.3-48) can be written in matrix form, yielding a tri-diagonal matrix of the form

$$\begin{bmatrix} \alpha_1+\beta_1 & -\beta_1 & 0 & 0 & \dots \\ -\beta_1 & \alpha_2+\beta_1+\beta_2 & -\beta_2 & 0 & \dots \\ 0 & -\beta_2 & \alpha_3+\beta_2+\beta_3 & -\beta_2 & \dots \\ & \ddots & & & \\ 0 & \dots 0 & -\beta_{\text{NC}} & \alpha_{\text{NSI}}+\beta_{\text{NC}}+\beta_{\text{NSI}} & -\beta_{\text{NSI}} \\ 0 & \dots\dots & 0 & -\beta_{\text{NSI}} & \alpha_{\text{NSO}}+\beta_{\text{NSI}}+\beta_{\text{NSO}} \end{bmatrix} \begin{bmatrix} T_2(1) \\ T_2(2) \\ T_2(3) \\ \\ \\ \\ T_2(\text{NSI}) \\ T_2(\text{NSO}) \end{bmatrix} = \begin{bmatrix} D_1 \\ D_2 \\ D_3 \\ \\ \\ \\ D_{\text{NSI}} \\ D_{\text{NSO}} \end{bmatrix} \tag{3.3-49}$$

where

$$\alpha_1 = \frac{M_{\text{f}}(i)\,\overline{c}_{\text{f}}(i)}{2\pi\Delta z(j)} \quad \text{for } i = 1,\dots \text{ NT} \tag{3.3-50}$$

$$\alpha_{\text{NE}''} = \frac{M_{\text{e}} c_{\text{e}}}{8\pi\Delta z(j)} \tag{3.3-51}$$

$$\alpha_{\text{NE}} = \frac{M_{\text{e}} c_{\text{e}}}{4\pi\Delta z(j)} = 2\alpha_{\text{NE}''} \tag{3.3-52}$$

$$\alpha_{\text{NE}'} = \alpha_{\text{NE}''} \tag{3.3-53}$$

$$\alpha_{\text{NC}} = \frac{\overline{\rho}_{\text{c}}(jc)\,\overline{c}_{\text{c}}(jc)\,A_{\text{c}}(jc)}{2\pi} + \frac{\overline{c}_{\text{c}}(jc)}{\pi\Delta z(jc)}\theta_2|w_2|\Delta t \tag{3.3-54}$$

$$\alpha_{\text{NSI}} = \frac{(\rho c)_{\text{sti}}\, d_{\text{sti}} S_{\text{pr}}}{2\pi} \tag{3.3-55}$$

$$\alpha_{\text{NSO}} = \frac{(\rho c)_{\text{sto}}\, d_{\text{sto}} S_{\text{pr}}}{2\pi} \tag{3.3-56}$$

$$\beta_{\text{i}} = \frac{r(i+1)\,\overline{k}_{\text{i,i+1}}}{\Delta r_{\text{i,i+1}}}\theta_2\,\Delta t \quad \text{for } i = 1,\dots NN \tag{3.3-57}$$

$$\beta_{\text{NT}} = r(\text{NR})\,\theta_2\,[h_b + \varepsilon\sigma h_{\text{r}}]\,\Delta t \tag{3.3-58}$$

$$\beta_{\text{NE}''} = \frac{\overline{r}_{\text{NE}}\,\overline{k}_{\text{NE}'',\text{NE}}}{\Delta r_{\text{NE}'',\text{NE}}}\theta_2\Delta t \tag{3.3-59}$$

$$\beta_{\text{NE}} = \frac{\overline{r}_{\text{NE}'}\,\overline{k}_{\text{NE,NE}'}}{\Delta r_{\text{NE,NE}'}}\theta_2\Delta t \tag{3.3-60}$$

$$\beta_{\text{NE}'} = r_{\text{NE}'}\,h_{c2}(j)\,\theta_2\Delta t \tag{3.3-61}$$

$$\beta_{\text{NC}} = \frac{S_{\text{pr}}}{2\,\pi} H_{\text{sic2}}(jc)\;\theta_2\Delta t \tag{3.3-62}$$

$$\beta_{\text{NSI}} = \frac{S_{\text{pr}}}{2\,\pi} H_{\text{stio}}(jc)\;\theta_2\Delta t \tag{3.3-63}$$

$$\beta_{\text{NSO}} = 0 \tag{3.3-64}$$

$$D_1 = T_1(1)\left[\alpha_1 - \frac{\theta_1}{\theta_2}\beta_1\right] + \frac{\theta_1}{\theta_2}\beta_1 T_1(2) + \psi_1 \tag{3.3-65}$$

$$\begin{aligned} D_{\mathrm{i}} = {} & \frac{\theta_1}{\theta_2}\beta_{\mathrm{i}-1} T_1(i-1) + T_1(i)\left[\alpha_1 - \frac{\theta_1}{\theta_2}(\beta_{\mathrm{i}-1} + \beta_{\mathrm{i}})\right] \\ & + \frac{\theta_1}{\theta_2}\beta_{\mathrm{i}} T_1(i+1) + \psi_1 \quad \text{for } i = 2, ..., NSI \end{aligned} \tag{3.3-66}$$

$$D_{\mathrm{NSO}} = \frac{\theta_1}{\theta_2}\beta_{\mathrm{NSI}} T_1(\mathrm{NSI}) + T_1(\mathrm{NSO})\left[\alpha_{\mathrm{NSO}} - \frac{\theta_1}{\theta_2}(\beta_{\mathrm{NSI}} + \beta_{\mathrm{NSO}})\right] + \psi_{\mathrm{NSO}} \tag{3.3-67}$$

$$\psi_{\mathrm{i}} = \frac{Q(i)\,\Delta t}{2\pi \Delta z(j)} \text{ for } i = 1, ..., \text{ NE}' \tag{3.3-68}$$

$$\psi_{\mathrm{NC}} = \frac{Q(\mathrm{NC})\,\Delta t}{2\pi \Delta z(j)} + \frac{\overline{c}_{\mathrm{c}}(jc)\,\Delta t}{\pi \Delta z(j)}\left[\theta_1 |w_1| T'_1 + \theta_2 |w_2| T'_2\right] \tag{3.3-69}$$

$$\psi_{\mathrm{NSI}} = \frac{Q(\mathrm{NSI})\,\Delta t}{2\pi \Delta z(j)} \tag{3.3-70}$$

and

$$\psi_{\mathrm{NSO}} = \frac{Q(\mathrm{NSO})\,\Delta t}{2\pi \Delta z(j)} + \frac{1}{2\pi} Q_{\mathrm{chch}} \Delta t \tag{3.3-71}$$

In these equations the α array is related to heat capacity, the β array is related to heat transfer between adjacent nodes, and the ψ array is related to the heat source.

The matrix equations 3.3-49 are solved by Gaussian elimination. First arrays A_i and S_i are defined:

$$A_{\mathrm{i}} = \alpha_{\mathrm{i}} + \beta_{\mathrm{i}} + \beta_{\mathrm{i}=1} - \frac{\beta^2_{\mathrm{i}-1}}{A_{\mathrm{i}-1}} \quad \text{for } i = 2, ..., \ NSO \tag{3.3-72}$$

$$S_1 = D_1 \tag{3.3-73}$$

and

$$S_{\mathrm{i}} = D_{\mathrm{i}} + \frac{\beta_{\mathrm{i}-1} S_{\mathrm{i}-1}}{A_{\mathrm{i}-1}} \quad \text{for } i = 2, ..., \ NSO \tag{3.3-74}$$

Then,

$$T_2(\mathrm{NSO}) = \frac{S_{\mathrm{NSO}}}{A_{\mathrm{NSO}}} \tag{3.3-75}$$

and

$$T_2(i) = S_{\mathrm{i}} + \frac{\beta_{\mathrm{i}}}{A_{\mathrm{i}}} T_2(i+1) \quad \text{for } i = NSI, \ NC, ..., \ 1 \tag{3.3-76}$$

3.3.2 Reflector Zones

In reflector zones, a two-node slab geometry treatment is used at each axial node for heat transfer to the "reflector". The reflector represents any material in the subassembly outside the pin section.

Typically, this material includes shield orifice blocks near the subassembly inlet, and instrumentation in the upper part of the subassembly. Usually, this material does not come to the form of either pure slabs or pure cylinders, and so any simple geometrical treatment of it will be only an approximation. The best that one is likely to do with a simple heat-transfer calculation is to use a slab calculation with parameters chosen to match total heat capacity, total heat-transfer surface area, and the approximate effective thickness of the material.

3.3.2.1 Basic Equations

The basic equations used for coolant and structure temperatures in reflector zones are the same as Eq. (3.3-5) and Eq. (3.3-10) used in the core and blanket regions, except that no heat generation is considered in the reflector zones. Therefore, the Q_c, term of Eq. (3.3-5) and the Q term of Eq. (3.3-10) are eliminated in the reflector zones. The reflector is treated with a one-dimensional heat conduction equation that is the same as the one used for the structure, except that the thermal properties ρ, c, and k used for the reflector can be different from those used for the structure.

3.3.2.2 Finite Difference Equations

Figure 3.2.6 shows the radial mesh structures used for an axial node in a reflector region.

3.3.2.2.1 Reflector Inner Node

Eq. (3.3-10) becomes

$$\begin{aligned}(\rho c)_{\mathrm{r}}\, d_{\mathrm{ri}} \frac{T_{\mathrm{ri2}}(jc) - T_{\mathrm{ri1}}(jc)}{\Delta t} = H_{\mathrm{rio}}(jc)\ \{\theta_1\ [T_{\mathrm{ro1}}(jc) - T_{\mathrm{ri1}}(jc)] \\ +\theta_2\,[T_{\mathrm{ro2}}(jc) - T_{\mathrm{ri2}}(jc)]\}\end{aligned} \tag{3.3-77}$$

where

ρc_{r} = density times specific heat of the reflector

d_{ri} = thickness of inner node

$T_{\mathrm{ri1}},\ T_{\mathrm{ri2}}$ = reflector inner node temperature at the beginning and end of the time step

$T_{\mathrm{ro}},\ T_{\mathrm{ro2}}$ = reflector outer node temperature at the beginning and end of the t ime step

d_{ro} = thickness of outer reflector node

and

$$H_{\mathrm{rio}} = \frac{2k_{\mathrm{r}}}{d_{\mathrm{ri}} + d_{\mathrm{ro}}} \tag{3.3-78}$$

Eq. (3.3-78) is obtained by adding thermal resistances in series:

$$\frac{1}{H_{\mathrm{rio}}} = \frac{d_{\mathrm{ri}}}{2k_{\mathrm{r}}} + \frac{d_{\mathrm{ro}}}{2k_{\mathrm{r}}} \tag{3.3-79}$$

Eq. (3.3-77) is similar to Eq. (3.3-48) for the outer structure node, except there is no direct heating source in the reflectors.

3.3.2.2.2 Reflector Outer Node

The equation for the reflector outer node temperature is similar to Eq. (3.3-46) for the structure inner node:

$$\begin{aligned}(\rho c)_{\mathrm{r}}\, d_{\mathrm{ro}} \frac{T_{\mathrm{ro2}}(jc) - T_{\mathrm{ro1}}(jc)}{\Delta t} \\ = \theta_1 H_{\mathrm{erc1}}(jc)\left[\overline{T}_{\mathrm{c1}}(jc) - T_{\mathrm{ro1}}(jc)\right] + \theta_2 H_{\mathrm{erc2}}(jc)\left[\overline{T}_{\mathrm{c2}}(jc) - T_{\mathrm{ro2}}(jc)\right] \\ +H_{\mathrm{rio}}(jc)\left\{\theta_1\left[T_{\mathrm{ri1}}(jc) - T_{\mathrm{ro1}}(jc)\right] + \theta_2\left[T_{\mathrm{ri2}}(jc) - T_{\mathrm{ro2}}(jc)\right]\right\}\end{aligned} \tag{3.3-80}$$

where

$$H_{\text{erc}} = \frac{h_{\text{c}} k_{\text{r}}}{k_{\text{r}} + \frac{h_{\text{c}} d_{\text{ro}}}{2}} \tag{3.3-81}$$

Note that

$$\overline{T}_{\text{c}}(jc) = [\, T_{\text{c}}(jc) + T_{\text{c}}(jc+1)] \;/\; 2 \tag{3.3-82}$$

3.3.2.2.3 Coolant Node

In the reflector zones and in the gas plenum, Eq. (3.3-5) becomes

$$\begin{aligned} \overline{\rho}_{\text{c}}(jc)\,\overline{c}_{\text{c}}(jc)\,\frac{\overline{T}_{\text{c2}}(jc) - T_{\text{c1}}(jc)}{\Delta t} + \frac{2\overline{c}_{\text{c}}(jc)}{\Delta z(jc)\,A_{\text{c}}(jc)} \{\theta_1 |w_1| \left[\overline{T}_{\text{c}}(jc) - T'_1\right] \\ + \theta_2 |w_2| \left[\overline{T}_{\text{c2}}(jc) - T'_2\right] \} \\ = \frac{S_{\text{er}}(\text{KZ})}{A_{\text{c}}(jc)} \{\theta_1 H_{\text{erc1}} \left[T_{\text{er1}}(jc) - T_{\text{c1}}(jc)\right] \\ + \theta_2 H_{\text{erc2}} \left[T_{\text{er2}}(jc) - \overline{T}_{\text{c2}}(jc)\right] \} + \frac{S_{\text{pr}}}{A_{\text{c}}(jc)} \{\theta_1 H_{\text{sic1}} \left[T_{\text{ri1}}(jc)\right. \\ \left. - T_{\text{c1}}(jc)\right] + \theta_2 H_{\text{sci2}}(jc) \left[T_{\text{sti2}}(jc) - \overline{T}_{\text{c2}}(jc)\right]\} \end{aligned} \tag{3.3-83}$$

where, in reflector zones,

$T_{\text{er1}} = T_{\text{ro1}}$

$T_{\text{er2}} = T_{\text{ro2}}$

$S_{\text{er}} = S_{\text{r}}$ = reflector perimeter

S_{pr} = structure perimeter

and

T_{sti} = structure inner node temperature

3.3.2.2.4 Structure Nodes

The finite difference approximation to Eq. (3.3-10) for the structure nodes is the same as in the core and blanket regions. Eq. (3.3-46) and Eq. (3.3-48) are used, except that Q_{st} is zero outside the core and blankets.

3.3.2.3 Solution of Finite Difference Equations

$$\begin{bmatrix} \alpha_1+\beta_1 & -\beta_1 & 0 & 0 & 0 \\ -\beta_1 & \alpha_2+\beta_1+\beta_2 & -\beta_2 & 0 & 0 \\ 0 & -\beta_2 & \alpha_3+\beta_2+\beta_3 & -\beta_3 & 0 \\ 0 & 0 & -\beta_3 & \alpha_4+\beta_3+\beta_4 & -\beta_4 \\ 0 & 0 & 0 & -\beta_4 & \alpha_5+\beta_4+\beta_5 \end{bmatrix} \begin{bmatrix} T_{ri2} \\ T_{ro2} \\ \overline{T}_{c2} \\ T_{Sti2} \\ T_{St02} \end{bmatrix} = \begin{bmatrix} D_1 \\ D_2 \\ D_3 \\ D_4 \\ D_5 \end{bmatrix} \tag{3.3-84}$$

where

$$\alpha_1 = (\rho c)_{\text{r}}\, d_{\text{ri}} S_{\text{er}}(\text{KZ}) \tag{3.3-85}$$

$$\alpha_2 = (\rho c)_{\text{r}}\, d_{\text{ro}} S_{\text{er}}\,(\text{KZ}) \tag{3.3-86}$$

$$\alpha_3 = \overline{\rho}_{\text{c}}\, \overline{c}_e\; A_{\text{c}} + 2\;\; \overline{c}_{\text{c}}\; \theta_2\, |w_2|\, \Delta t \tag{3.3-87}$$

$$\alpha_4 = (\rho c)_{\text{sti}}\, d_{\text{sti}} S_{\text{pr}} \tag{3.3-88}$$

$$\alpha_5 = (\rho c)_{\text{sto}}\, d_{\text{sto}} S_{\text{pr}} \tag{3.3-89}$$

$$\beta_1 = \theta_2 H_{\text{rio}}\,(jc)\, S_{\text{er}} \Delta t \tag{3.3-90}$$

$$\beta_2 = \theta_2 H_{\text{erc2}}\,(jc)\, S_{\text{er}} \Delta t \tag{3.3-91}$$

$$\beta_3 = \theta_2 H_{\text{sic2}}\,(jc)\, S_{\text{pr}} \Delta t \tag{3.3-92}$$

$$\beta_4 = \theta_2 H_{\text{stio}}\,(jc)\, S_{\text{pr}} \Delta t \tag{3.3-93}$$

$$\beta_5 = \theta \tag{3.3-94}$$

$$D_1 = T_{\text{ri1}} \left[\alpha_1 - \frac{\theta_1}{\theta_2}\beta_1\right] + \frac{\theta_1}{\theta_2}\, \beta_1\, T_{\text{ro1}} \tag{3.3-95}$$

$$\begin{aligned} D_{\text{i}} = \frac{\theta_1}{\theta_2}\, \beta_{\text{i}-1} T''_1\,(\text{i}-1) + T''_1\,(i) \left[\alpha_1 - \frac{\theta_1}{\theta_2}\,(\beta_{\text{i}-1} + \beta_1)\right] \\ + \frac{\theta_1}{\theta_2}\beta_{\text{i}} T''_1\,(i+1) \quad \text{for } i = 2, 4 \end{aligned} \tag{3.3-96}$$

$$\begin{aligned} D_3 = \frac{\theta_1}{\theta_2}\beta_2 T''_1\,(2) + T''_1\,(3) \left[\alpha_3 - \frac{\theta_1}{\theta_2}\,(\beta_2 + \beta_3)\right] + \frac{\theta_1}{\theta_2}\beta_3 T''_1\,(4) \\ + 2\frac{\overline{c}_c \Delta t}{\Delta z}\left[\theta_1\, |w_1|\, T'_1 + \theta_2\, |w_2|\, T'_2\right] \end{aligned} \tag{3.3-97}$$

$$D_5 = \frac{\theta_1}{\theta_2}\beta_4 T''_1\,(4) + T''_1\,(5) \left[\alpha_5 - \frac{\theta_1}{\theta_2}\,(\beta_4 + \beta_5)\right] + S_{\text{pr}}\, Q_{\text{chch}} \Delta t \tag{3.3-98}$$

$$T''_1\,(1) \;=\; T_{\text{ri1}}\,(jc) \tag{3.3-99}$$

$$T''_1\,(2) \;=\; T_{\text{ro1}}\,(jc) \tag{3.3-100}$$

$$T''_1\,(3) \;=\; \overline{T}_{\text{c1}}\,(jc) \tag{3.3-101}$$

$$T''_1\,(4) \;=\; T_{\text{sti1}}\,(jc) \tag{3.3-102}$$

and

$$T''_1\,(5) \;=\; T_{\text{sto1}}\,(jc) \tag{3.3-103}$$

This tri-diagonal matrix is solved in the same manner as in Section 3.3.1.2 above.

3.3.3 Gas Plenum Region

In the gas plenum region, a single gas node is in contact with all axial cladding nodes. A single radial node is used in the cladding, as indicated in Figure 3.2.5.

3.3.3.1 Basic Equations

The basic equations used in the cladding, coolant, and structure in the gas plenum region are the same as those used in the core and blankets, except that there is no heat source term outside of the core and blankets. The gas is assumed to transfer heat only to the cladding, and the basic equation used for the gas temperature is

$$(z_{\text{pt}} - z_{\text{pb}})\,(\rho c)_{\text{g}} A_{\text{g}} \frac{\text{dT}_{\text{g}}}{\text{dt}} = 2\pi r_{\text{brp}} H_{\text{eg}} \int_{z_{\text{pb}}}^{z_{\text{pt}}} (T_{\text{e}} - T_{\text{g}})\,\text{dz} \tag{3.3-104}$$

where

z_{pb} = elevation at the bottom of the gas plenum

z_{pt} = elevation at the top of the gas plenum

$(\rho c)_{\text{g}}$ = density times specific heat for the plenum gas

A_{g} = cross sectional area of the gas plenum

$$A_{\text{g}} = \pi r_{\text{brp}}^2 \tag{3.3-105}$$

r_{brp} = cladding inner radius in the gas plenum region

T_{g} = gas temperature

H_{eg} = heat-transfer coefficient from the plenum gas to the cladding node

T_{e} = cladding temperature

3.3.3.2 Finite Difference Equations

3.3.3.2.1 Gas Plenum

Eq. (3.3-105) becomes

$$\begin{aligned} &(\rho c)_{\text{g}}\, A_{\text{g}} \frac{(T_{\text{g2}} - T_{\text{g1}})}{\Delta t} \\ &= 2\pi r_{\text{brp}} H_{\text{eg}} \frac{\sum_{\text{jp}} \{\theta_1 [T_{\text{e1}}(jp) - T_{\text{g1}}] + \theta_2 [T_{\text{e2}}(jp) - T_{\text{g2}}]\}\, \Delta z(jp)}{\sum_{\text{jp}} \Delta z(jp)} \end{aligned} \tag{3.3-106}$$

where

jp = plenum node

T_{g1} = plenum gas temperature at the beginning of the time step

T_{g2} = plenum gas temperature at the end of the time step

$$H_{\text{eg}} = \frac{1}{R_{\text{g}} + \frac{r_{\text{erp}} - r_{\text{brp}}}{2k_{\text{ep}}}} = \frac{2k_{\text{ep}}}{r_{\text{erp}} - r_{\text{brp}} + 2k_{\text{ep}} R_{\text{g}}} \tag{3.3-107}$$

k_{ep} = cladding thermal conductivity in the gas plenum

and

R_{g} = thermal resistance of the gas.

3.3.3.2.2 Cladding Node

The equation for the cladding node is

$$\begin{aligned}\rho_{\mathrm{e}} c_{\mathrm{e}} A_{\mathrm{ep}} \frac{T_{\mathrm{e2}}(jp) - T_{\mathrm{e1}}(jp)}{\Delta t} \\ = 2\pi r_{\mathrm{erp}} \left\{ \theta_1 \, H_{\mathrm{erc1}} \left[\overline{T}_{\mathrm{c1}}(jc) - T_{\mathrm{e1}}(jp) \right] \right. \\ \left. + \theta_2 H_{\mathrm{erc2}} \left[\overline{T}_{\mathrm{c2}}(jc) - T_{\mathrm{e2}}(jp) \right] \right\} + \; 2\pi r_{\mathrm{brp}} H_{\mathrm{eg}} \left\{ \theta_1 \left[T_{\mathrm{g1}} - T_{\mathrm{e1}}(jp) \right] \right. \\ \left. + \theta_2 \left[T_{\mathrm{g2}} - T_{\mathrm{e2}}(jp) \right] \right\}\end{aligned} \tag{3.3-108}$$

where

ρ_{e} = cladding density

c_{e} = cladding specific heat

$$A_{\mathrm{ep}} = \pi \left[r_{\mathrm{erp}}^2 - r_{\mathrm{brp}}^2 \right] \tag{3.3-109}$$

and

r_{erp} = cladding outer radius in the gas plenum region

3.3.3.2.3 Coolant Node

The equation for the coolant node in the gas plenum region is the same as Eq. (3.3-83) for reflector zones, except that in the gas plenum

$$H_{\mathrm{erc}} = \frac{h_{\mathrm{e}} k_{\mathrm{e}}}{k_{\mathrm{e}} + \frac{h_{\mathrm{c}} (r_{\mathrm{erp}} - r_{\mathrm{brp}})}{2}} \tag{3.3-110}$$

$$S_{\mathrm{cr}} = 2\pi r_{\mathrm{erp}} \tag{3.3-111}$$

and

k_{e} = cladding thermal conductivity.

3.3.3.2.4 Structure Nodes

The finite difference approximations used for the structure in the gas plenum region are the same as those used in the core and blankets. Eq. (3.3-46) and Eq. (3.3-48) are used, except that Q_{st} is zero.

3.3.3.3 Solution of Finite Difference Equations

A direct solution of the finite difference equations would be complicated, since Eq. (3.3-106) connects a number of axial nodes, each containing a number of radial nodes. Instead, an approximate solution method is used. This method uses the assumption that the total heat capacity of the gas is much less than the total heat capacity of the cladding in the gas plenum region, or

$$(\rho c)_{\mathrm{g}} \; A_{\mathrm{g}} \; < < \; \rho_{\mathrm{e}} \; c_{\mathrm{e}} \; A_{\mathrm{ep}} \tag{3.3-112}$$

which should always be the case.

The solution method contains five steps:

1. Set $T_{g2} = T_{g1}$.
2. Solve Eq. (3.3-108), Eq. (3.3-83), Eq. (3.3-46), and Eq. (3.3-48) for all axial nodes in the gas plenum to get new cladding, coolant, and structure temperatures.
3. Use Eq. (3.3-106) to obtain a new computed value for T_{g2}.
4. For each axial node, calculate the heat flow error, ΔE, due to the assumption in step 1:

$$\Delta E = (\rho c)_{g} \; A_{g} \; (T_{g2} - T_{g1}) \tag{3.3-113}$$

1. Add this heat flow to the cladding, changing the cladding temperature:

$$T_{e2} = T_{c2} + \frac{\Delta E}{\rho_{e} c_{e} A_{ep}} \tag{3.3-114}$$

In step 2, the equations for each axial node give a matrix equation of the form

$$\begin{bmatrix} \alpha_1 + \beta_1 & -\beta_1 & 0 & 0 \\ -\beta_1 & \alpha_1 + \beta_1 + \beta_2 & -\beta_2 & 0 \\ 0 & -\beta_2 & \alpha_3 + \beta_2 + \beta_3 & -\beta_3 \\ 0 & 0 & -\beta_3 & \alpha_4 + \beta_3 + \beta_4 \end{bmatrix} \begin{bmatrix} T''_2(1) \\ T''_2(2) \\ T''_2(3) \\ T''_2(4) \end{bmatrix} = \begin{bmatrix} D_1 \\ D_2 \\ D_3 \\ D_4 \end{bmatrix} \tag{3.3-115}$$

where

$$T''_2(1) = T_{e2}(jp) \tag{3.3-116}$$

$$T''_2(2) = \overline{T}_{c2}(jc) \tag{3.3-117}$$

$$T''_2(3) = T_{sti2}(jc) \tag{3.3-118}$$

$$T''_2(4) = T_{sto2}(jc) \tag{3.3-119}$$

$$\alpha_1 = \rho_{e} c_{e} A_{ep} + 2\pi r_{brp} \Delta t \theta_2 \; H_{eg} \tag{3.3-120}$$

$$\alpha_2 = \overline{\rho}_{e} \overline{c}_{e} A_{e} + 2\overline{c}_{c} \theta_2 \; |w_2| \, \Delta t \tag{3.3-121}$$

$$\alpha_3 = (\rho c)_{sti} \; d_{sti} \; S_{pr} \tag{3.3-122}$$

$$\alpha_4 = (\rho c)_{sto} \, d_{sto} \; S_{pr} \tag{3.3-123}$$

$$\beta_1 = 2\pi \; r_{erp} \theta_2 \; H_{erc2} \; (jc) \, \Delta t \tag{3.3-124}$$

$$\beta_2 = \theta_2 \; H_{sic2} \; (jc) \, S_{pr} \Delta t \tag{3.3-125}$$

$$\beta_3 = \theta_2 \; H_{stio2} \; (jc) \, S_{pr} \Delta t \tag{3.3-126}$$

$$\beta_4 = 0 \tag{3.3-127}$$

$$D_1 = T''_1(1) \; \left[\alpha_1 - \frac{\theta_1}{\theta_2} \; \beta_1 \right] + \frac{\theta_1}{\theta_2} \; \beta_1 \; T''_1(2) + \; 2\pi r_{brp} \Delta t H_{eg} \left[T_{g1} - T_{e1}(jp) \right] \tag{3.3-128}$$

$$\begin{aligned} D_2 = \frac{\theta_1}{\theta_2} \beta_1 \; T''_1(1) + T''_1 \; (2) \; \left[\alpha_2 - \frac{\theta_1}{\theta_2} \; (\beta_1 + \beta_2) \right] + \frac{\theta_1}{\theta_2} \; \beta_2 \; T''_1(3) \\ + 2 \frac{\overline{c}_{c} \Delta t}{\Delta z} \left[\theta_1 \; |w_1| \, T'_1 + \theta_2 \, |w_2| \, T'_2 \right] \end{aligned} \tag{3.3-129}$$

$$D_3 = \frac{\theta_1}{\theta_2} \beta_2 \; T''_1(2) + T''_1(3) \; \left[\alpha_1 - \frac{\theta_1}{\theta_2} \; (\beta_2 + \beta_3) \right] + \frac{\theta_1}{\theta_2} \; \beta_3 \; T''_1(4) \tag{3.3-130}$$

$$D_4 = \frac{\theta_1}{\theta_2}\beta_3\, T''_1\,(3) + T''_1\,(4)\left[\alpha_4 - \frac{\theta_1}{\theta_2}\,(\beta_3+\beta_4)\right] + S_{\text{pr}}\, Q_{chch}\Delta t \tag{3.3-131}$$

$$T''_1\,(1) = T_{\text{e1}}\,(jp) \tag{3.3-132}$$

$$T''_1\,(2) = \overline{T}_{\text{c1}}\,(jc) \tag{3.3-133}$$

$$T''_1\,(3) = T_{\text{sti1}}\,(jc) \tag{3.3-134}$$

and

$$T''_1\,(4) = T_{\text{sto1}}\,(jc) \tag{3.3-135}$$

This tri-diagonal matrix equation is solved in the same manner as in Section 3.3.1.2 above. For step 3, Eq. (3.3-106) is rewritten as

$$T_{g2} = \frac{T_{g1} + d'\, s'}{1 + d'\,\theta_2} \tag{3.3-136}$$

where

$$d' = \frac{2\pi\, r_{\text{brp}}\, H_{\text{eg}}\Delta t}{(\rho c)_g\, A_g} \tag{3.3-137}$$

and

$$s' = \frac{\sum_{\text{jp}} \{\theta_1\,(T_{\text{c1}}\,(jp) - T_{\text{g1}}) + \theta_2\, T_{\text{e2}}\,(jp)\}\,\Delta z\,(jp)}{\sum_{\text{jp}} \Delta z\,(jp)} \tag{3.3-138}$$

3.3.4 Order of Solution

For a time step, the coolant flow rates are calculated before any temperatures are calculated. Extrapolated coolant temperatures are used to obtain temperature-dependent coolant properties for the flow-rate calculations. Section 3.9 describes the pre-voiding coolant flow-rate calculations.

The order in which the temperature calculations are carried out depends on whether the coolant flow is up or down. In either case, the calculation goes in the direction of the flow. For upward flow, the calculation starts at the subassembly inlet. The coolant temperature at the first coolant node is set equal to the inlet temperatures, T_{in}:

$$T_{\text{c2}}\,(1) = T_{\text{in}} \tag{3.3-139}$$

The inlet temperature is supplied by the primary loop calculation, as discussed in Chapter 5. For each axial coolant node, jc, the coolant temperature, $T_{\text{c2}}(jc)$, at the bottom of the node is used as input for the simultaneous calculation of temperatures at all radial nodes. The solutions in Section 3.3.1, *3.3.2*, and *3.3.3* provide $\overline{T}_{\text{c2}}\,(jc)$, the average coolant temperature for the axial node. The $\overline{T}_{\text{c2}}\,(jc+1)$ is obtained from

$$T_{c2}\,(jc+1) = 2\overline{T}_{\text{c2}}\,(jc) - T_{\text{c2}}\,(jc) \tag{3.3-140}$$

This coolant temperature at the top of the node jc is the coolant inlet temperature used as input for the calculation at node $jc+1$.

If the flow is downward, the process is the same, except that the calculation starts at the top of the subassembly and works down. The coolant reentry temperatures, $T_{\text{up}\updownarrow}$, described in Section 3.3.6 is used as the starting point for the last coolant node MZC:

$$T_{\text{c2}}\,(\text{MZC}) = T_{\text{upl}} \tag{3.3-141}$$

For each coolant node, jc, $T_{\text{c2}}(jc+1)$ is used as input to the calculation of all radial nodes. The solutions in Section 3.3.1, *3.3.2*, and *3.3.3* again provide $\overline{T}_{\text{c2}}\,(jc)$. Then $T_{\text{c2}}\,(jc)$ is obtained from

$$T_{\text{c2}}\,(jc) = 2\overline{T}_{\text{c2}}\,(jc) - T_{\text{c2}}\,(jc+1) \tag{3.3-142}$$

3.3.5 Melting of Fuel or Cladding

As discussed in Section 3.3.1.1, melting of both fuel and cladding is treated with a melting range form a solidus temperature, T_{sol}, to a liquidus temperature, T_{liq}, rather than using a sharp melting temperature. Between the solidus and the liquidus, an effective specific heat, c_{m}, based on the heat of fusion is used, as in Eq. (3.3-3). In the actual temperature calculations for a time step, the heat of fusion is neglected. Then, at the end of the step the temperatures are modified to account for the heat of fusion at any radial node that is going through the melting range. A number of different cases are considered, depending on the relationships between T_1, the temperature at the beginning of the step, and T_2, the temperature calculated for the end of the step, ignoring melting, T_{sol} and T_{liq}. In all cases, T''_2 is the final adjusted temperature at the end of the step, and T'_2 would be the final adjusted temperature if it did not go beyond the melting range.

Case 1: $T_1 < T_{\text{sol}}, T_2 > T_{\text{sol}}$

In this case, an adjusted temperature,T'_2, is calculated as

$$T'_2 = T_{\text{sol}} + (T_2 - T_{\text{sol}}) \frac{c}{c_{\text{m}}} \tag{3.3-143}$$

where c is the normal specific heat that was used in the calculation of T_2. Then,

$$T'_2 = \begin{cases} T_{\text{liq}} + (T'_2 - T_{\text{liq}}) \frac{c_{\text{m}}}{c} & \text{if } T'_2 > T_{\text{liq}} \\ T'_2 & \text{otherwise} \end{cases} \tag{3.3-144}$$

In essence, Eq. (3.3-143) divides the energy above the solidus by c_{m} to get the temperature above the solidus; and Eq. (3.3-144) divides any energy above the liquidus by c to get the temperature above the liquidus.

Case 2: $T_1 > T_{\text{liq}}, T_2 < T_{\text{liq}}$

In this case,

$$T'_2 = T_{\text{liq}} + (T_2 - T_{\text{liq}}) \frac{c}{c_{\text{m}}} \tag{3.3-145}$$

$$T'_2 = \begin{cases} T_{\text{sol}} + (T'_2 - T_{\text{sol}}) \frac{c_{\text{m}}}{c} & \text{if } T'_2 > T_{\text{sol}} \\ T'_2 & \text{otherwise} \end{cases} \tag{3.3-146}$$

Case 3: $T_{\text{liq}} \geq T_1 \geq T_{\text{sol}}$

In this case

$$T'_2 = T_1 + (T_2 - T_1) \frac{c}{c_{\text{m}}} \tag{3.3-147}$$

$$T'_2 = \begin{cases} T_{\text{liq}} + (T'_2 - T_{\text{liq}}) \frac{c_m}{c} & \text{if } T'_2 > T_{\text{liq}} \\ T_{\text{sol}} + (T'_2 - T_{\text{sol}}) \frac{c_m}{c} & \text{if } T'_2 < T_{\text{sol}} \\ T'_2 & \text{otherwise} \end{cases} \tag{3.3-148}$$

3.3.6 Coolant Inlet and Re-entry Temperature

The coolant inlet temperature at the subassembly inlet and the re-entry temperature at the subassembly outlet are determined mainly by the coolant plenum temperatures calculated in PRIMAR-4; but, in addition, mixing zones are modeled at the inlet and outlet of each channel. Therefore, the temperature of coolant entering a subassembly is based on the temperature in the part of the plenum in the immediate vicinity of the end of the subassembly, and the reentry temperature soon after an expulsion is based on both the bulk plenum temperature and the temperature of the coolant that was recently expelled.

The mixing mode uses a mass, M_{mix}, of the coolant in the mixing volume, a specific heat, C_{mix}, of the coolant, and a time constant, τ_{mix}, for heat transfer or mixing between the mixing volume and the bulk plenum coolant. If all temperatures and flows are constant, then the mixing volume temperature, T_{mix}, is assumed to approach an equilibrium value asymptotically with an exponential decay. Figure 3.3.1 shows the model for the outlet mixing volume.

If T_{mix1} is the mixing volume temperature at the beginning of a time step, the T_{mix2} is the value at the end, then

$$T_{\text{mix2}} = T_{\text{eq}} + (T_{\text{mix1}} - T_{\text{eq}})\, e^{-\frac{\Delta t}{\tau}} \tag{3.3-149}$$

For the outlet plenum,

$$\frac{1}{\tau} = \frac{1}{\tau_{\text{mix}}} + \frac{|\overline{w}|}{M_{\text{mix}}} \tag{3.3-150}$$

where $\overline{w}$ is the average coolant flow rate at the subassembly outlet. The equilibrium temperature is

$$T_{\text{eq}} = \begin{cases} T_{\text{out}} & \text{if } \overline{w} \leq 0 \\ \left(\frac{T_{\text{out}}}{\tau_{\text{mix}}} + \frac{\overline{w}\overline{T}_{\text{exp}}}{M_{\text{mix}}} \right) \tau & \text{if } \overline{w} > 0 \end{cases} \tag{3.3-151}$$

where $\overline{T}_{\text{exp}}$ is the average temperature of the sodium expelled from the top of the subassembly into the mixing volume during the time step. T_{out} is the bulk temperature in the outlet plenum. For the inlet plenum, the same equations are used except that $\overline{w}$ changes sign because positive flow results in flow from the inlet mixing volume into the subassembly inlet. For the inlet, Eq. (3.3-151) and Eq. (3.3-151) become

$$T_{\text{eq}} = \begin{cases} T_{\text{in}} & \text{if } \overline{w} \leq 0 \\ \left(\frac{T_{\text{in}}}{\tau_{\text{mix}}} - \frac{\overline{w}\overline{T}_{\text{exp}}}{M_{\text{mix}}} \right) \tau & \text{if } \overline{w} > 0 \end{cases} \tag{3.3-152}$$

where T_{in} is the bulk temperature in the inlet plenum.

The coolant inlet and reentry temperature calculations described above are used in both the pre-voiding module and the boiling module except when a vapor bubble has blown out of the top of the channel. In this case, the condensation of vapor in the outlet plenum raises the temperature in the mixing volume. A two-step process is used for each time step. In the first step, vapor condensation raises the mixing volume temperature. In the second step, heat transfer or mixing with the bulk plenum coolant is accounted for. For the first step, a condensation heat-transfer time constant, τ_{c}, is calculated as

$$\tau_{\text{c}} = \frac{M_{\text{mix}}\, C_{\text{mix}}}{h_{\text{cond}} \Delta Z_{\text{v}}\, p_{\text{er}}} \tag{3.3-153}$$

where h_{cond} is the vapor condensation heat-transfer coefficient, ΔZ_{v} is the length of the vapor bubble beyond the subassembly outlet, and p_{er} is the bubble perimeter or surface area per unit length. The value of p_{er} is taken from the coolant channel dimensions in the top node below the subassembly outlet. For step 1, the new mixing volume temperature, T_{new}, is calculated as

$$T_{\text{new}} = \overline{T}_{\text{v}} + \left(T_{mix1} - \overline{T}_{\text{v}}\right) e^{-\frac{\Delta t}{\tau_{\text{c}}}} \tag{3.3-154}$$

where $\overline{T}_{v}$ is the average vapor temperature in the part of the bubble above the subassembly outlet. In the second step, the final mixing volume temperature is calculated as

$$T_{\text{mix2}} = T_{\text{new}} + (T_{\text{out}} - T_{\text{new}}) \left(1 - e^{-\frac{\Delta t}{\tau_{\text{mix}}}}\right) \tag{3.3-155}$$

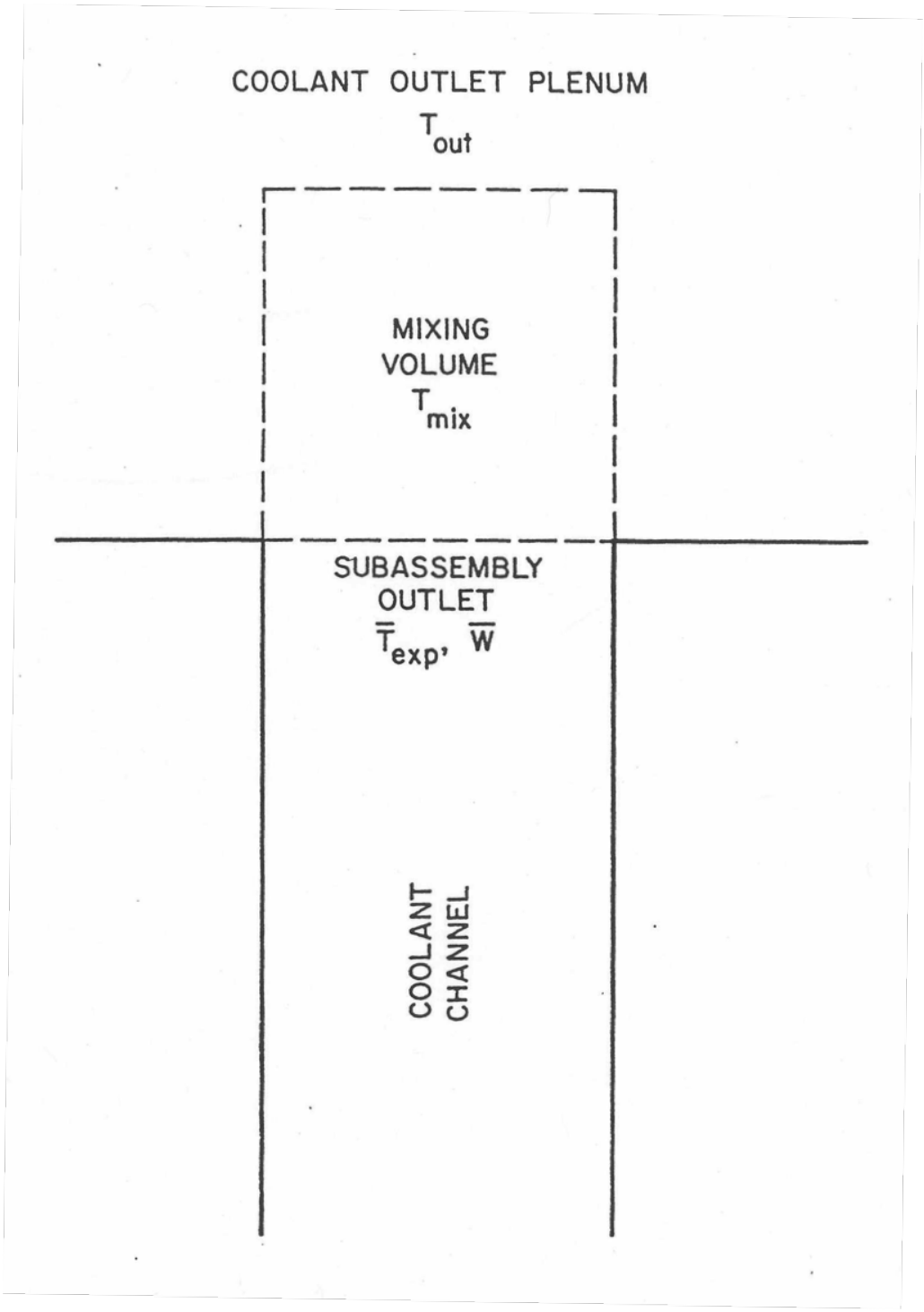


Figure 3.3.1: Coolant Re-entry Temperature Model

3.4 Steady-State Thermal Hydraulics

The steady-state thermal hydraulics calculations for a channel using the single pin per subassembly option consist of direct solutions of the relevant steady-state equations, rather than running the transient calculations until they converge to a steady-state solution.

For the steady-sate calculations, the user specifies the coolant flow rate for each channel, the coolant inlet temperature and exit pressure, and the power in each node of each channel. The code then calculates the remaining coolant temperatures and pressures, as well as the temperatures in the fuel, cladding, structure, and reflectors. First, the coolant temperatures in a channel are calculated, starting at the inlet and working upward. The steady-state coolant temperature calculation requires only the coolant flow rate, the total power in each axial node, and the coolant heat capacity; so coolant temperatures can be calculated before the fuel and cladding temperatures are known. The second step is to calculate the coolant pressures, starting at the subassembly outlet and working down. Inlet orifice coefficients are adjusted so that all channels have the same total pressure drop. The pressure calculations are described in Section 3.9. The third step is to set the structure and reflector temperatures equal to the coolant temperatures everywhere except in the core and axial blankets. The gas plenum temperatures are also set equal to the coolant temperature in this region, and the cladding temperature in the gas plenum region is also set equal to the coolant temperature. Fourth, the fuel-pin temperatures are calculated for each axial node in the core and axial blankets, starting at the cladding outer surface and working inward. Last, the structure temperatures in the core and axial blankets are calculated.

3.4.1 Basic Equations

The basic heat-transfer equations used in the steady-state calculations are the same as those used for the transient solution, except that all of the time derivatives are dropped in the steady-state solution. These equations include Eq. (3.3-1) and Eq. (3.4-4) to Eq. (3.3-10). Also, the spatial finite differencing used in the steady-state is the same as that used in the transient.

For the steady-state calculations, Eq. (3.3-5) becomes

$$\frac{\mathrm{d}}{\mathrm{dz}}\left(wc_{\mathrm{c}}T\right)=Q_{\mathrm{ct}}A_{\mathrm{c}} \tag{3.4-1}$$

where the total heat source per unit volume, Q_{ct}, at node jc is

$$Q_{\mathrm{ct}}\left(jc\right)=Q_{\mathrm{c}}\left(jc\right)+Q_{\mathrm{ec}}\left(jc\right)+Q_{\mathrm{sc}}\left(jc\right)=\frac{\overline{P}\left(jc\right)}{A_{\mathrm{c}}}\Delta z\left(jc\right) \tag{3.4-2}$$

and $\overline{P}\left(jc\right)$ is the total steady-state power (watts) in the node. For this equation, it is assumed that all heat generated in the fuel, cladding, and structure ends up in the coolant. Note that outside the core and axial blankets $\overline{P}\left(jc\right)$ and $Q_{\mathrm{ct}}\left(jc\right)$ are zero.

For the steady-state fuel and cladding calculations, Eq. (3.3-1) is multiplied by $2\pi r$ and integrated from the fuel inner surface, r_{if}, to give

$$2\pi kr\frac{\mathrm{dT}}{\mathrm{dr}}=-2\pi\int_{r_{\mathrm{if}}}^{r}r'Q\left(r'\right)dr' \tag{3.4-3}$$

where the adiabatic boundary condition at r_{if} has been used.

3.4.2 Coolant Temperatures

The finite difference form for Eq. (3.4-1) may be written as

$$w\overline{c}_{\mathrm{c}}\,(\mathrm{j})\,\frac{[T_c\,(jc+1)-T_c\,(jc)]}{\Delta z\,(jc)}=\frac{\overline{P}\,(jc)}{\Delta z\,(jc)} \tag{3.4-4}$$

or

$$T_{\mathrm{c}}\,(jc+1)=T_{\mathrm{c}}\,(jc)+\frac{\overline{P}\,(jc)}{w\overline{c}_{\mathrm{c}}\,(jc)} \tag{3.4-5}$$

where $\overline{c}_{\mathrm{c}}\,(jc)$ is the specific heat evaluated at the average temperature, $\overline{T}_{\mathrm{c}}\,(jc)$, given by

$$\overline{T}_{\mathrm{c}}\,(jc)=\frac{T_{\mathrm{c}}\,(jc)+T_{\mathrm{c}}\,(jc+1)}{2} \tag{3.4-6}$$

Also, $T_{\mathrm{c}}\,(1)$ is equal to the inlet temperature:

$$T_{\mathrm{c}}\,(1)=T_{\mathrm{in}} \tag{3.4-7}$$

Starting from *jc* = 1, Eq. (3.4-5) is used to match up the channel. An iteration is used to obtain consistency between $\overline{c}_{\mathrm{c}}\,(jc)$ and $\overline{T}_{\mathrm{c}}\,(jc)$.

3.4.3 Fuel and Cladding Temperatures in the Core and Axial Blankets

At each axial node, the radial node powers, $Q(i)$, are calculated using Eq. (3.3-23) to Eq. (3.3-26). Note that the Q in Eq. (3.4-3) is a power per unit volume, whereas $Q(i)$ is an integral value for a node:

$$Q\,(i)=\int_{z(j)}^{z(j+1)}\int_{r_{\mathrm{i}}}^{r_{i+1}}2\pi r\;\;Qdr\;\;\mathrm{dz} \tag{3.4-8}$$

The sums, $Q_{\mathrm{sm}}(i)$, are calculated as

$$Q_{\mathrm{sm}}\,(i)=\sum_{ii=1}^{i}\mathrm{Q}\,(ii) \tag{3.4-9}$$

Eq. (3.4-3) becomes

$$2\pi\;\overline{k}_{\mathrm{i,i+1}}\;\frac{r\,(i+1)\,[T\,(i+1)-T\,(i)]}{\Delta r_{\mathrm{i,i+1}}}\;=\;\frac{Q_{\mathrm{sm}}\,(i)}{\Delta z} \tag{3.4-10}$$

or

$$T\,(i)=T\,(i+1)+\frac{\Delta r_{\mathrm{i,i+1}}Q_{\mathrm{sm}}\,(i)}{2\pi k_{\mathrm{i,i+1}}r\,(i+1)\,\Delta z} \tag{3.4-11}$$

where $\Delta r_{i,i+1}$ and $\overline{k}_{i,i+1}$ are given by Eq. (3.3-22) and Eq. (3.3-27).

The calculations for an axial node start with the coolant temperature that has already been calculated, as in the section above:

$$T\,(\mathrm{NC},j)=\overline{T}\,(jc) \tag{3.4-12}$$

Then the cladding surface temperature is given by

$$T\,(\mathrm{NE}')=T\,(\mathrm{NC})+\frac{Q_{\mathrm{sm}}\,(\mathrm{NE}')}{2\pi r\,(\mathrm{NE}')\,\Delta z\;h_{\mathrm{c}}} \tag{3.4-13}$$

Cladding temperatures at nodes NE and NE″ are calculated using Eq. (3.4-11). Since $\overline{k}_{\mathrm{ii+1}}$ can be a function of T_{i}, a simple iteration between Eq. (3.4-11) and Eq. (3.3-27) is used.

The equation used for the fuel surface temperature is

$$2\pi r\,(\mathrm{NR})\left\{h_{\mathrm{b}}\left[T\,(\mathrm{NT})-T\,(\mathrm{NE}'')\,\right]+\varepsilon\sigma\left[T\,(\mathrm{NT})^{4}-T\,(\mathrm{NE}'')^{4}\right]\right\}=\frac{Q_{\mathrm{sm}}\,(\mathrm{NT})}{\Delta z} \tag{3.4-14}$$

or

$$T\,(\mathrm{NT})=d_1-d_2T\,(\mathrm{NT})^{4} \tag{3.4-15}$$

where

$$d_1=T\,(\mathrm{NE}'')+\frac{Q_{\mathrm{sm}}\,(\mathrm{NT})}{2\pi r\,(\mathrm{NR})\,\Delta z\;h_{\mathrm{b}}}+\frac{\epsilon\sigma T\,(\mathrm{NE}'')^{4}}{h_{\mathrm{b}}} \tag{3.4-16}$$

and

$$d_2=\frac{\epsilon\sigma}{h_{\mathrm{b}}} \tag{3.4-17}$$

Eq. (3.4-15) is solved by iteration.

After the fuel surface temperature has been calculated, the inner fuel node temperatures are calculated one at a time, starting at the outside and working inward, by iterating between Eq. (3.4-11) and Eq. (3.3-27). In this procedure, *T*(*i*) is to be found after *T*(*i*+1) is known. First, *T*(*i*) is set equal to $T(i+1)$. Second, *k*(*i*) is to be found after *T*(*i*+1) is known. First, *T*(*i*) is set equal to $T(i+1)$. Second, *k*(*i*) is calculated as a function of the temperature, *T*(*i*). Third, $\overline{k}_{\mathrm{i,i+1}}$ is calculated using Eq. (3.3-27). Fourth, a new value for *T*(*i*) is calculated, using Eq. (3.4-11). Fifth, the new *T*(*i*) from the fourth step is compared with old value used in the second step. If the two values differ by less than a user-specified convergence criterion, then the iteration is finished, and the code goes on to the next node. Otherwise, the code goes back to the second step, using the new value of *T*(*i*), and repeats the process.

3.4.4 Structure Temperatures in the Core Axial Blankets

The inner structure node temperature is calculated using

$$\Delta z\;S_{\mathrm{pr}}\;H_{\mathrm{sic}}\;\left[T\,(\mathrm{NSI})-T\,(\mathrm{NC})\right]=\gamma_{\mathrm{s}}\overline{P}\,(j) \tag{3.4-18}$$

or

$$T\,(\mathrm{NSI})=T\,(\mathrm{NC})+\frac{\gamma_{\mathrm{s}}\overline{P}\,(j)}{\Delta z\;S_{\mathrm{pr}}\;H_{\mathrm{sic}}} \tag{3.4-19}$$

The outer structure node is then calculated using

$$T\,(\mathrm{NSO})=T\,(\mathrm{NSI})+\frac{\gamma_{\mathrm{s}}\overline{P}\,(j)\;\;d_{\mathrm{sto}}}{\Delta z\;S_{\mathrm{pr}}\;H_{\mathrm{stio}}\;\;(d_{\mathrm{sti}}+d_{\mathrm{sto}})} \tag{3.4-20}$$

3.4.5 Reflector, Structure, Cladding, and Gas Plenum Temperature Outside the Core and Axial Blankets

Outside the core and axial blankets no power sources are considered, so the reflector and structure temperatures at an axial node are the same as the coolant temperature for the steady-state. The coolant temperatures are the same at all axial nodes in the gas plenum region, and the cladding and gas temperatures in this region are equal to the coolant temperatures.

3.5 Transient Heat Transfer after the Start of Boiling

After the start of boiling, the coolant temperatures are calculated in the coolant routines, rather than being calculated simultaneously with fuel, cladding, and structure temperatures. Coupling between the boiling calculations and the non-coolant heat-transfer calculation takes place in two parts for each time step. First, the boiling routines us extrapolated cladding and structure temperatures to calculate the heat fluxes to the coolant for the boiling calculation. Then the heat fluxes actually used in the coolant routines are passed to the heat-transfer routines to be used as boundary conditions at the cladding, structure, and reflector surfaces. The net results of this procedure are that energy is conserved, and a fully implicit boiling calculation can be made without requiring a direct simultaneous solution of all of the fuel-pin temperatures in the boiling model. The coupling through extrapolated cladding and structure temperatures and heat fluxes at the cladding and structure surfaces imposes numerical stability limitations on the heat-transfer time-step sizes. Currently, fuel-pin temperatures are calculated at the end of the fuel-pin heat-transfer time step, whereas structure and reflector temperatures are calculated at every coolant time step. The coolant time step can be no longer than the heat-transfer step, and the coolant step is often much shorter. For typical fuel pins, the stability limit for the heat-transfer time step is of the order of .02 s. With a thin structure, the stability limit for structure temperature calculations could be less than .02 s, although for typical duct wall thicknesses (.12 in. or .003 m) the stability limit would be closer to one second. Should timing studies indicate that the structure and reflector temperature calculations account for a significant fraction of the total computing time, then the code will be modified so as to do these calculations less often than once every coolant time step.

3.5.1 Fuel and Cladding Temperatures in the Core and Axial Blanket

The equations used for fuel and cladding temperatures after the switch to the boiling module are the same as those used in the non-voiding module, except that in the boiling module the fuel-pin heat-transfer calculations stop at the cladding outer surface rather than carrying through to the structure outer node. The finite difference equations for radial nodes 1-NE are the same as Eq. (3.3-17) to Eq. (3.3-37). For node NE′ , the cladding outer node, the heat flux to the coolant at the cladding surface must be accounted for. Also, in a boiling region, a film of liquid sodium can be left on the cladding. Since the film is in intimate contact with the cladding, the heat capacity of the film is added to the heat capacity of the cladding outer node, rather than being accounted for in the boiling calculation. Thus, the finite difference equation for node NE′ becomes

$$\left[\frac{m_{\mathrm{e}}c_{\mathrm{e}}}{4}+2\pi r\left(\mathrm{NE}'\right)\rho_{\mathrm{c}}c_{\mathrm{c}}w_{\mathrm{fe}}\Delta z\left(j\right)\right]\cdot\left[\frac{T_2\left(\mathrm{NE}'\right)-T_1\left(\mathrm{NE}'\right)}{\Delta t}\right]$$
$$=-\frac{2\pi\overline{r}_{\mathrm{NE}'}\Delta z\left(j\right)\ \overline{k}_{\mathrm{NE,NE}'}}{\Delta r_{\mathrm{NE,NE}'}}\left\{\theta_{21}\left[T_1\ \left(\mathrm{NE}'\right)-T_1\left(\mathrm{NE}\right)\right]\right.$$
$$\left.+\theta_2\ \left[T_2\ \left(\mathrm{NE}'\right)-T_2\left(\mathrm{NE}\right)\right]\right\}-2\pi r\left(\mathrm{NE}'\right)\Delta z\left(j\right)\frac{E_{\mathrm{ec}}\left(j\right)}{\Delta t}$$
$$+Q\left(\mathrm{NE}'\right)\quad(3.5\text{-}1)$$

where w_{fe} is the thickness of liquid sodium film left on the cladding after voiding occurs, and E_{ec} is the integrated heat flux from cladding to coolant.

The value of E_{ec} is computed in the coolant routines as

$$E_{\mathrm{ec}}\left(jc\right)=\int_t^{t+\Delta t}\frac{T_{\mathrm{eex}}\left(jc\right)-\overline{T}_c\left(jc\right)}{R_{\mathrm{ec}}\left(jc\right)}\,\mathrm{dt}'\qquad(3.5\text{-}2)$$

where

T_{eex} = extrapolated cladding temperature at a point ¼ of the way from the outer cladding surface to the inner cladding surface:

$$T_{\text{eex}}(j,t') = f_1 \frac{[\text{T}(\text{NE},j,t_1) + T(\text{NEP},j,t_1)]}{2} + f_2 \frac{[\text{T}(\text{NE},j,t_2) + T(\text{NEP},j,t_2)]}{2} \tag{3.5-3}$$

$$f_1 = \frac{t_2 - t'}{t_2 - t_1} \tag{3.5-4}$$

$$f_1 = 1 - f_1 \tag{3.5-5}$$

$$R_{\text{ec}} = \frac{1}{h_{\text{c}}} + R_{\text{ehf}} \tag{3.5-6}$$

and

$$R_{\text{ehf}} = \frac{[r(\text{NE}') - r(\text{NE})]}{k_{\text{e}}} \left\{ \frac{r(\text{NE}') \left[1 - \frac{\gamma_{\text{e}}}{4(1-\gamma_{\text{c}}-\gamma_{\text{s}})}\right]}{r(\text{NE}) + 3r(\text{NE}')} \right\} \tag{3.5-7}$$

If a node is partly voided, then E_{ec} is averaged over the length of the node as well as integrated over time.

The $\rho_{\text{c}} c_{\text{c}} w_{\text{fe}}$ term is supplied by the boiling routines. It represents the heat capacity of any liquid film left on the cladding after voiding occurs. This term is zero unless voiding has occurred at this axial node. Since the film temperature tends to follow the cladding surface temperature more closely than it follows the vapor temperature, the film heat capacity is lumped with the cladding outer node, and film temperatures are not explicitly calculated in the boiling routines.

The finite difference equations are again put in a matrix form like Eq. (3.3-49), except that in the voiding case there are only NE′ elements. The definitions of α, β, and D are the same as in Eq. (3.3-50) to 3.3-51 for nodes 1-NE. For node NE′, α is still given by Eq.

$$\beta_{\text{NE}'} = 0 \tag{3.5-8}$$

Eq. (3.3-53) or Eq. (3.3-51), except that the $\rho_{\text{c}} c_{\text{c}} w_{\text{fe}}$ term is added to it. Also,

and

$$D_{\text{NE}'} = \frac{\theta_1}{\theta_2} \beta_{\text{NE}} T_1(\text{NE}) + T_1(\text{NE}') \left[\alpha_{\text{NE}'} - \frac{\theta_1}{\theta_2} \beta_{\text{NE}} \right] + \psi_{\text{NE}'} - r(\text{NE}') E_{\text{ec}}(j) \tag{3.5-9}$$

The equations are solved in the same manner as in the non-voiding case.

3.5.2 Structure Temperatures

The basic equations used for the two structure radial nodes at an axial node are

$$(\rho c)_{\text{sto}} d_{\text{sto}} \frac{\text{dT}_{\text{sto}}}{\text{dt}} = H_{\text{stio}} (T_{\text{sti}} - T_{\text{sto}}) + Q_{\text{st}} f_{\text{o}} \tag{3.5-10}$$

and

$$\left[(\rho c)_{\text{sti}} d_{\text{sti}} + \rho_{\text{c}} c_{\text{c}} w_{\text{fst}} \right] \frac{\text{dT}_{\text{sti}}}{\text{dt}} = H_{\text{stio}} (T_{\text{sto}} - T_{\text{sti}}) + Q_{\text{st}} f_{\text{i}} + \frac{(T_{\text{c}} - T_{\text{sti}})}{R_{\text{sc}}} \tag{3.5-11}$$

where

$$f_{\mathrm{i}} = \frac{d_{\mathrm{sti}}}{d_{\mathrm{sti}} + d_{\mathrm{sto}}} \tag{3.5-12}$$

$$f_{\mathrm{o}} = \frac{d_{\mathrm{sto}}}{d_{\mathrm{sti}} + d_{\mathrm{sto}}} \tag{3.5-13}$$

and w_{fst} = thickness of the liquid-sodium film left on the structure after voiding occurs.

The heat capacity of the film in the boiling region, as represented by the $\rho_{\mathrm{c}} c_{\mathrm{c}} w_{\mathrm{fst}}$ term, is supplied by the boiling routines to be added to the inner structure node.

The structure temperature calculation is either a semi-implicit or a fully implicit calculation, depending on the time-step in relation to an inner structure node heat-transfer time constant, τ_{sti}, calculated as

$$\tau_{\mathrm{sti}} = \frac{(\rho c)_{\mathrm{sti}}\, d_{\mathrm{sti}}^2}{2k_{\mathrm{sti}}} \tag{3.5-14}$$

If Δt is less than τ_{sti} then the semi-implicit calculation is used. Otherwise the fully implicit calculation is used.

3.5.2.1 Semi-Implicit Calculations

For the semi-implicit calculation, finite differencing of Eq. (3.5-10) and Eq. (3.5-11) gives

$$(\rho c)_{\mathrm{sto}}\, d_{\mathrm{sto}} \frac{(T_{\mathrm{sto2}} - T_{\mathrm{sto1}})}{\Delta t} = \frac{H_{\mathrm{stio}}}{2} \left[T_{\mathrm{sti2}} - T_{\mathrm{sto2}} + T_{\mathrm{sti1}} - T_{\mathrm{sto1}}\right] + Q_{\mathrm{st}} f_{\mathrm{o}} \tag{3.5-15}$$

and

$$\left[(\rho c)_{\mathrm{sti}}\, d_{\mathrm{sti}} + \rho_{\mathrm{c}} c_{\mathrm{c}} w_{\mathrm{fst}}\right] \frac{(T_{\mathrm{sti2}} - T_{\mathrm{sti1}})}{\Delta t} = \frac{H_{\mathrm{stio}}}{2} \left[T_{\mathrm{sto2}} - T_{\mathrm{sti2}} + T_{\mathrm{sto1}} - T_{\mathrm{sti1}}\right] + Q_{\mathrm{st}}\, f_{\mathrm{i}} - \frac{E_{\mathrm{sc}}}{\Delta t} \tag{3.5-16}$$

where

$$E_{\mathrm{sc}}\,(jc) = \int_{\mathrm{t}}^{t+\Delta t} \frac{T_{\mathrm{stex}}\,(jc) \; - \overline{T}_{\mathrm{c}}\,(jc)}{R_{\mathrm{sc}}\,(jc)}\, \mathrm{dt}' \tag{3.5-17}$$

The value of E_{sc} is computed in the boiling routines. T_{stex} is the extrapolated structure inner node temperature, and

$$R_{\mathrm{sc}} = \frac{1}{h_{\mathrm{c}}} + \frac{d_{\mathrm{sti}}}{2k_{\mathrm{sti}}} \tag{3.5-18}$$

Eq. (3.5-15) and Eq. (3.5.2.1) are put in the form

$$a_{11} T_{\mathrm{sto2}} + a_{12} T_{\mathrm{sti2}} = b_1 \tag{3.5-19}$$

$$a_{21} T_{\mathrm{sto2}} + a_{22} T_{\mathrm{sti2}} = b_2 \tag{3.5-20}$$

with the solutions

$$T_{\mathrm{sto2}} = \frac{a_{22} b_1 - a_{12} b_2}{a_{11} a_{22} - a_{12} a_{21}} \tag{3.5-21}$$

and

$$T_{\mathrm{sti2}} = \frac{a_{11} b_2 - a_{21} b_2}{a_{11} a_{22} - a_{12} a_{21}} \tag{3.5-22}$$

where

$$a_{11} = (\rho c)_{\text{sto}}\ d_{\text{sto}} + \frac{\Delta t}{2} H_{\text{stio}} \tag{3.5-23}$$

$$a_{12} = -\frac{\Delta t}{2}\ H_{\text{stio}} \tag{3.5-24}$$

$$a_{21} = a_{12} \tag{3.5-25}$$

$$a_{22} = (\rho c)_{\text{sti}}\, d_{\text{sti}} + \frac{\Delta t}{2} H_{\text{stio}} + \rho_{\text{c}} c_{\text{c}} w_{\text{fst}} \tag{3.5-26}$$

$$b_1 = \left[(\rho c)_{\text{sto}}\, d_{\text{sto}} - \frac{\Delta t}{2} H_{\text{stio}} \right] T_{\text{sto1}} + Q_{\text{st}} f_0 \Delta t - a_{12} T_{\text{sti1}} \tag{3.5-27}$$

and

$$b_2 = \left[(\rho c)_{\text{sti}}\, d_{\text{stio}} - \frac{\Delta t}{2}\ H_{\text{stio}} + \rho_{\text{c}} c_{\text{c}} w_{\text{fst}} \right] T_{\text{sti1}} - a_{12}\ T_{\text{sto1}} + Q_{\text{st}} f_{\text{i}} \Delta t - \Delta t Q_{\text{sc}} \tag{3.5-28}$$

3.5.2.2 Fully Implicit Calculations

Since the inner structure node may represent only a small fraction of the total structure thickness, τ_{sti} can be small. If the time-step size is appreciably larger than τ_{sti} then the semi-implicit calculation can become numerically unstable. Therefore, a different algorithm is used for larger time-step sizes. This algorithm uses two steps. First, a fully implicit calculation is made, using a coolant temperature and thermal resistance to the coolant as structure surface boundary conditions, rather than using the integrated heat flux. In this first step, the heat flux from the coolant to the structure will, in general, not match the heat flux form structure to coolant used in the coolant calculations. Therefore, in the second step, the inner node and outer node structure temperatures are both adjusted by the same amount so that the integrated heat flux from structure to coolant is matched.

For the first step the finite difference equations used for the two structure node temperatures are

$$(\rho c)_{\text{sto}}\, d_{\text{sto}} \frac{(T_{\text{sto2}} - T_{\text{sto1}})}{\Delta t} = H_{\text{stio}}\, (T_{\text{sti2}} - T_{\text{sto2}}) + Q_{\text{st}} f_o \tag{3.5-29}$$

and

$$\left[(\rho c)_{\text{sti}}\, d_{\text{sti}} + \rho_{\text{c}} c_{\text{c}} w_{\text{fst}} \right] \frac{(T_{\text{sti2}} - T_{\text{sti1}})}{\Delta t} = H_{\text{stio}}\, (T_{\text{sto2}} - T_{\text{sti2}}) + \frac{T_{\text{c}} - T_{\text{sti2}}}{R_{\text{sc}}} + Q_{\text{st}} f_{\text{i}} \tag{3.5-30}$$

where the values for T_{c} and R_{sc} are supplied by the coolant routines. The solutions for these two equations again have the same form as Eq. (3.5-21) and Eq. (3.5-22) except that in this case the coefficients are defined as

$$a_{11} = (\rho c)_{\text{sto}}\, d_{\text{sto}} + \Delta t H_{\text{stio}} \tag{3.5-31}$$

$$a_{12} = a_{21} = -\Delta t H_{\text{stio}} \tag{3.5-32}$$

and

$$a_{22} = (\rho c)_{\text{sti}}\, d_{\text{sti}} + \rho_{\text{c}} c_{\text{c}} w_{\text{fst}} + \Delta t H_{\text{stio}} + \frac{\Delta t}{R_{\text{sc}}} \tag{3.5-33}$$

The temperature difference, ΔT_{st}, between the outer and inner nodes is then defined as

$$\Delta T_{\text{st}} = T_{\text{sto2}} - T_{\text{sti2}} \tag{3.5-34}$$

In the second step, ΔT_{st} is preserved but the temperatures are adjusted so as to match the value of E_{sc} supplied by the coolant routines:

$$\left[(\rho c)_{sti}\, d_{sti} + \rho_c c_c w_{fst}\right](T_{sci2} - T_{sti1}) + (\rho c)_{sto}\, d_{sto}\,(T_{sto2} - T_{sto1}) = Q_{st} - E_{sc} \tag{3.5-35}$$

The solution to Eq. (3.5-34) and Eq. (3.5-35) is

$$\begin{aligned} T_{sto2} = \{&Q_{st}\Delta t - E_{sc} + (\rho c)_{sto}\, d_{sto} T_{sto1} \\ &+ \left[(\rho c)_{sti}\ d_{sti} + \rho_c c_c w_{fst}\right] \cdot (T_{sti1} + \Delta T_{st})\} \left[(\rho c)_{sto}\ d_{sto} + (\rho c)_{sti}\, d_{sti} + \rho_c c_c w_{fst}\right] \end{aligned} \tag{3.5-36}$$

and

$$T_{sti2} = T_{sto2} - \Delta T_{st} \tag{3.5-37}$$

Note that the second step can still cause numerical instabilities if the time-step size is too large or the total structure thickness is too small, but in the fully implicit scheme the stability limit is based on the total structure thickness, whereas in the semi-implicit scheme the stability is based mainly on the inner node thickness.

3.5.3 Reflector Temperatures

In the boiling module the treatment of reflector temperatures is almost identical to the structure temperature treatment. The main difference is that in the reflector, the outer node is in contact with the coolant, whereas in the structure, the inner code is in contact with the coolant. Also, in the reflector the density, specific heat, and thermal conductivity are the same for both nodes, whereas in the structure these properties can vary from inner node to outer node.

The basic equations are

$$(\rho c)_r\, d_{ri} \frac{dT_{ri}}{dt} = H_{rio}\,(T_{ro} - T_{ri}) \tag{3.5-38}$$

and

$$\left[(\rho c)_r\, d_{ro} + \rho_c c_c w_{fr}\right] \frac{dT_{ro}}{dt} = H_{rio}\,(T_{ri} - T_{ro}) + (T_c - T_{ro})\, H_{erc} \tag{3.5-39}$$

where

w_{fr} = thickness of the liquid-sodium film left on the reflector after voiding occurs.

An outer reflector node heat-transfer time constant, τ_{ro}, is calculated as

$$\tau_{ro} = \frac{(\rho c)_r\, d_{ro}^2}{2k_r} \tag{3.5-40}$$

and a fully implicit calculation is used if the time-step size is greater than τ_{ro}. Otherwise a semi-implicit calculation is used.

3.5.3.1 Semi-Implicit Calculations

Finite differencing of Eq. (3.5-38) and Eq. (3.5-39) gives

$$(\rho c)_r\, d_{ri} \frac{(T_{ri2} - T_{ri1})}{\Delta t} = \frac{H_{rio}}{2}\,(T_{ro2} - T_{ri2} + T_{ro1} - T_{ri1}) \tag{3.5-41}$$

$$\left[(\rho c)_r\, d_{ro} + \rho_c c_c w_{fr}\right] \frac{(T_{ri2} - T_{ri1})}{\Delta t} = \frac{H_{rio}}{2}\,(T_{ri2} - T_{ro2} + T_{ri1} - T_{ro1}) - \frac{E_{rc}}{\Delta t} \tag{3.5-42}$$

where

$$E_{\text{rc}} = \int_t^{t+\Delta t} \left(T_{\text{rex}} - \overline{T}_{\text{c}}\right) H_{\text{erc}} \text{dt}' \tag{3.5-43}$$

These equations are put in the form

$$a_{11} T_{\text{ri2}} + a_{12} T_{\text{ro2}} = b_1 \tag{3.5-44}$$

$$a_{21} T_{\text{ri2}} + a_{22} T_{\text{ro2}} = b_2 \tag{3.5-45}$$

with the solution

$$T_{\text{ri2}} = \frac{a_{22} b_1 - a_{12} b_2}{a_{11} a_{22} - a_{12} a_{21}} \tag{3.5-46}$$

and

$$T_{\text{ro2}} = \frac{a_{11} b_2 - a_{21} b_1}{a_{11} a_{22} - a_{12} a_{21}} \tag{3.5-47}$$

The coefficients are

$$a_{11} = (\rho c)_{\text{r}} d_{\text{ri}} + \frac{\Delta t}{2} H_{\text{rio}} \tag{3.5-48}$$

$$a_{12} = -\frac{\Delta t}{2} H_{\text{rio}} \tag{3.5-49}$$

$$a_{21} = a_{12} \tag{3.5-50}$$

$$a_{22} = (\rho c)_{\text{r}} d_{\text{ro}} + \frac{\Delta t}{2} H_{\text{rio}} + \rho_{\text{c}} c_{\text{c}} w_{\text{fr}} \tag{3.5-51}$$

$$b_1 = \left[(\rho c)_{\text{r}} d_{\text{ri}} - \frac{\Delta t}{2} H_{\text{rio}}\right] T_{\text{ri1}} + \frac{\Delta t}{2} H_{\text{rio}} T_{\text{ro1}} \tag{3.5-52}$$

and

$$b_2 = \left[(\rho c)_{\text{r}} d_{\text{ro}} + \rho_{\text{c}} c_{\text{c}} w_{\text{fr}} - \frac{\Delta t}{2} H_{\text{rio}}\right] T_{\text{ro1}} + \frac{\Delta t}{2} H_{\text{rio}} T_{\text{ri1}} - E_{\text{rc}} \tag{3.5-53}$$

3.5.3.2 Fully Implicit Calculations

As in the structure temperature case, a two-step process is used. In the first step, the finite difference equations used are

$$(\rho c)_{\text{r}} d_{\text{ri}} \frac{(T_{\text{ri2}} - T_{\text{ri1}})}{\Delta t} = H_{\text{rio}} \left(T_{\text{ro2}} - T_{\text{ri2}}\right) \tag{3.5-54}$$

and

$$\left[(\rho c)_{\text{r}} d_{\text{ro}} + \rho_{\text{c}} c_{\text{c}} w_{\text{fr}}\right] \frac{(T_{\text{ro2}} - T_{\text{ro1}})}{\Delta t} = H_{\text{rio}} \left(T_{\text{ri2}} - T_{\text{ro2}}\right) + \left(\overline{T}_{\text{c}} - T_{\text{ro2}}\right) H_{\text{erc}} \tag{3.5-55}$$

The solutions again have the same form as Eq. (3.5-46) and Eq. (3.5-47), with the coefficients given by

$$a_{11} = (\rho c)_{\text{r}} d_{\text{ri}} + \Delta t H_{\text{rio}} \tag{3.5-56}$$

$$a_{12} = -\Delta t H_{\text{rio}} \tag{3.5-57}$$

$$a_{21} = a_{12} \tag{3.5-58}$$

$$a_{22} = (\rho c)_{\mathrm{r}}\, d_{\mathrm{ro}} + \Delta t H_{\mathrm{rio}} + \rho_{\mathrm{c}} c_{\mathrm{c}} w_{\mathrm{fr}} \tag{3.5-59}$$

$$b_1 = (\rho c)_{\mathrm{r}}\, d_{\mathrm{ri}} T_{\mathrm{ri1}} \tag{3.5-60}$$

and

$$b_2 = \left[(\rho c)_{\mathrm{r}}\, d_{\mathrm{ro}} + \rho_{\mathrm{c}} c_{\mathrm{c}} w_{\mathrm{fr}}\right] T_{\mathrm{rol}} + \overline{T}_{\mathrm{c}} \Delta t\; H_{\mathrm{erc}} \tag{3.5-61}$$

The temperature difference between nodes, ΔT_{r}, is defined as

$$\Delta T_{\mathrm{r}} = T_{\mathrm{ri2}} - T_{\mathrm{ro2}} \tag{3.5-62}$$

In the second step, ΔT_{r} is preserved and E_{rc} is matched. The energy conservation equation is

$$(\rho c)_{\mathrm{r}}\, d_{\mathrm{ri}} \left(T_{\mathrm{ri2}} - T_{\mathrm{ri1}}\right) + \left[(\rho c)_{\mathrm{r}}\, d_{\mathrm{ro}} + \rho_{\mathrm{c}} c_{\mathrm{c}} w_{\mathrm{fr}}\right] \left(T_{\mathrm{ro2}} - T_{\mathrm{ro1}}\right) = -E_{\mathrm{rc}} \tag{3.5-63}$$

The solution to Eq. (3.5-62) and Eq. (3.5-63) is

$$T_{\mathrm{ri2}} = \frac{\left\{-E_{\mathrm{rc}} + (\rho c)_{\mathrm{r}}\, d_{\mathrm{ri}} T_{\mathrm{ri1}} + \left[(\rho c)_{r}\, d_{\mathrm{ro}} + p_{\mathrm{c}} c_{\mathrm{c}} w_{\mathrm{fr}}\right] \cdot \left(T_{\mathrm{rol}} + \Delta T_{\mathrm{r}}\right)\right\}}{\left[(\rho c)_{\mathrm{r}} \left(d_{\mathrm{ri}} + d_{\mathrm{ro}}\right) + p_{\mathrm{c}} c_{\mathrm{c}} w_{\mathrm{fr}}\right]} \tag{3.5-64}$$

and

$$T_{\mathrm{ro2}} = T_{\mathrm{ri2}} - \Delta T_{\mathrm{r}} \tag{3.5-65}$$

3.5.4 Gas Plenum Region

The basic equations used for the cladding and gas temperatures in the gas plenum region are Eq. (3.5-68) and the following equation:

$$\rho_{\mathrm{e}} c_{\mathrm{e}} A_{\mathrm{ep}} \frac{\mathrm{dT_e}\left(jp\right)}{\mathrm{dt}} = 2\pi r_{\mathrm{erp}} H_{\mathrm{erc}} \left[\overline{T}_{\mathrm{c}}\left(jc\right) - T_{\mathrm{e}}\left(jp\right)\right] + 2\pi r_{\mathrm{brp}} H_{\mathrm{eg}} \left[T_{\mathrm{g}} - T_{\mathrm{e}}\left(jp\right)\right] \tag{3.5-66}$$

Since Eq. (3.5-68) links all of the cladding nodes in the gas plenum, a direct semi-implicit or implicit solution of Eq. (3.5-68) and Eq. (3.5-66) would require a simultaneous solution for the gas temperature and all of the cladding node temperatures. Instead, the cladding temperatures are calculated first, using the gas temperature at the beginning of the time step. Then the gas temperature is calculated using the newly computed cladding temperatures.

Finite differencing of Eq. (3.5-66) gives

$$\rho_{\mathrm{e}} c_{\mathrm{e}} A_{\mathrm{ep}} \frac{\left(T_{\mathrm{e2}} - T_{\mathrm{e1}}\right)}{\Delta t} = -2\pi r_{\mathrm{erp}} \frac{E_{\mathrm{ec}}}{\Delta t} + \pi r_{\mathrm{brp}} H_{\mathrm{eg}} \left(2T_{\mathrm{g1}} - T_{e2} - T_{\mathrm{e1}}\right) \tag{3.5-67}$$

where E_{ec} is calculated in the coolant routines in the same manner as indicated in Eq. (3.5-2), except that in the gas plenum only one radial node is used in the cladding, and R_{ehf} becomes

$$R_{\mathrm{ehf}} = \frac{r_{\mathrm{erp}} - r_{\mathrm{brp}}}{2k_{\mathrm{e}}} \tag{3.5-68}$$

The solution of Eq. (3.5-67) for T_{e2} gives

$$T_{\mathrm{e2}} = \frac{\left\{\left(\rho_{\mathrm{e}} c_{\mathrm{e}} A_{\mathrm{ep}} - \pi r_{\mathrm{brp}} H_{\mathrm{eg}} \Delta t\right) T_{\mathrm{el}} - 2\pi r_{\mathrm{erp}} E_{\mathrm{cc}} + 2\pi r_{\mathrm{brp}} H_{\mathrm{eg}} \Delta t T_{\mathrm{g1}}\right\}}{\left(\rho_{\mathrm{e}} c_{\mathrm{e}} A_{\mathrm{ep}} + \pi r_{\mathrm{brp}} H_{\mathrm{eg}} \Delta t\right)} \tag{3.5-69}$$

In the second step, Eq. (3.5-70) is used with $\theta_1 = \theta_2 = 1/2$. The solution for T_{g2} is

$$T_{g2} = \frac{\left[(\rho c)_g A_g - \pi r_{brp} H_{eg} \Delta t\right] T_{g1} + \pi r_{brp} H_{eg} \Delta t \frac{s_1}{s_2}}{(\rho c)_g A_g + \pi r_{brp} H_{eg} \Delta t} \tag{3.5-70}$$

where

$$s_1 = \sum_{jp} \left[T_{e1}(jp) + T_{e2}(jp)\right] \Delta z(jp) \tag{3.5-71}$$

and

$$s_2 = \sum_{jp} \Delta z(jp) \tag{3.5-72}$$

3.5.5 Coolant Temperatures in Liquid Slugs

Before the onset of coolant voiding, coolant temperatures are calculated at all node boundaries, as indicated in Figure 3.2.3. After the start of boiling, liquid coolant temperatures are calculated at all node boundaries outside vapor regions, as well as at moving nodes near the bubble interfaces. Two different types of calculations are made. Eulerian temperature calculations are made for fixed coolant nodes in the inlet and outlet liquid slugs. Lagrangian temperature calculations are made for the moving interface nodes and for any fixed nodes in liquid slugs between bubbles. There is also an option to use Lagrangian temperature calculations for all nodes, both fixed and moving.

The Eulerian calculation is probably more accurate for the fixed nodes. The main disadvantage of this method is that a sudden jump in inlet temperature can lead to a sawtooth temperature pattern, with the temperature high at one node, low at the next, and high again at the third node. The Lagrangian calculation does not exhibit this behavior. This sawtooth behavior is not unstable: the perturbation at any node is no larger than the jump in the inlet temperature, and the perturbations tend to die out in later time steps. Also, the coolant inlet and reentry temperature calculations described in Section 3.3.6 tend to eliminate sudden jumps in inlet and reentry temperatures.

3.5.5.1 Eulerian Temperature Calculation

The basic equation used in this calculation is again Eq. (3.3-5). The heat fluxes Q_{ec} and Q_{sc} are calculated as

$$Q_{ec} = \frac{\left(T_e - \overline{T}_c\right)}{R_{ec}} \frac{2\pi r\,(\mathrm{NE}')}{A_c} \tag{3.5-73}$$

and

$$Q_{sc} = \frac{\left(T_{si} - \overline{T}_c\right)}{R_{sc}} \frac{S_{st}}{A_c} \tag{3.5-74}$$

where R_{ec} and R_{sc} are given by Eq. (3.5-6) and Eq. (3.5-18), T_e is the average of $T(\mathrm{NE})$ and $T(\mathrm{NE}')$, and T_{si} is the inner structure node temperature. In reflector zones, T_e is replaced by the reflector outer node temperature; and in the gas plenum region, the one radial cladding node temperature is used. In the boiling module, the coolant temperatures are calculated before the cladding and structure temperatures are, so linear extrapolation in time is used to obtain values of T_e and T_{si} at the end of a time step.

A semi-implicit finite differencing of Eq. (3.3-5) gives

$$\begin{aligned}\overline{p}\,(jc)\,\overline{c}_{\mathrm{c}}\,(jc)\,A_{\mathrm{c}}\,(jc)\,\frac{[T_{\mathrm{c2}}\,(jc+1)+T_{\mathrm{c2}}\,(jc)-T_{\mathrm{c1}}\,(jc+1)+T_{\mathrm{c1}}\,(jc)]}{2\Delta t}\\ +\overline{c}_{\mathrm{c}}\,(jc)\,w_1\frac{[T_{\mathrm{c1}}\,(jc+1)+T_{\mathrm{c1}}\,(jc)]}{2\Delta z\,(jc)}+\overline{c}_{\mathrm{c}}\,(jc)\,w_2\frac{[T_{\mathrm{c2}}\,(jc+1)-T_{\mathrm{c2}}\,(jc)]}{2\Delta z\,(jc)}\\ =Q_{\mathrm{c}}\,(jc)\,A_{\mathrm{c}}\,(jc)+\frac{k_5\,(jc)\,A_{\mathrm{c}}\,(jc)}{4}\left\{\frac{2T_{\mathrm{e2}}\,(jc)-T_{\mathrm{e2}}\,(jc)-T_{\mathrm{c2}}\,(jc+1)}{R_{\mathrm{ec2}}\,(jc)}\right.\\ +\frac{2T_{\mathrm{e1}}\,(jc)-T_{\mathrm{e1}}\,(jc)-T_{\mathrm{c1}}\,(jc+1)}{R_{\mathrm{ec1}}\,(jc)}+\gamma_2\,(jc)\left[\frac{2T_{\mathrm{si2}}\,(jc)-T_{\mathrm{c2}}\,(jc)-T_{\mathrm{c2}}\,(jc+1)}{R_{\mathrm{sc2}}\,(jc)}\right.\\ \left.\left.+\frac{2T_{\mathrm{si1}}\,(jc)-T_{\mathrm{c1}}\,(jc)-T_{\mathrm{c1}}\,(jc+1)}{R_{\mathrm{sc1}}\,(jc)}\right]\right\}\end{aligned}\tag{3.5-75}$$

where

$$k_5\,(jc)=\begin{cases}\frac{2\pi r\left(\mathrm{NE}',jc\right)}{A_{\mathrm{c}}(jc)} & \text{in the core and blankets}\\ \frac{s_{\mathrm{er}}(\mathrm{kz})}{A_{\mathrm{c}}(jc)} & \text{in a reflector region}\\ \frac{2\pi r_{\mathrm{erp}}}{A_{\mathrm{c}}(jc)} & \text{in the gas plenum region}\end{cases}\tag{3.5-76}$$

and

$$\gamma_2\,(jc)=\frac{S_{\mathrm{st}}\,(jc)}{k_5\,(jc)\,A_{\mathrm{c}}\,(jc)}\tag{3.5-77}$$

Solving for $T_{\mathrm{c2}}\,(jc+1)$ gives

$$\begin{aligned}T_{\mathrm{c2}}\,(jc+1)=\left\{T_{\mathrm{c1}}\,(jc+1)\left[\overline{\rho}_{\mathrm{c}}\,(jc)-\frac{\Delta t w_1}{\Delta z\,(jc)\,A_{\mathrm{c}}\,(jc)}-\frac{k_5\,(jc)\,\Delta t h_{\mathrm{br}}\,(jc)}{2\overline{c}_{\mathrm{c}}\,(jc)}\right]\right.\\ +T_{\mathrm{c2}}\,(jc)\left[-\rho\,(jc)+\frac{\Delta t w_2}{\Delta z\,(jc)\,A_{\mathrm{c}}\,(jc)}-\frac{k_5\,(jc)\,\Delta t}{2\overline{c}_{\mathrm{c}}\,(jc)}h_{\mathrm{b2}}\,(jc)\right]\\ +T_{\mathrm{c1}}\,(jc)\left[\rho_{\mathrm{c}}\,(jc)+\frac{\Delta t w_1}{\Delta z\,(jc)\,A_{\mathrm{c}}\,(jc)}-\frac{k_5\,(jc)\,\Delta t}{2\overline{c}_{\mathrm{c}}\,(jc)}h_{\mathrm{b1}}\,(jc)\right]\\ \left.+\frac{4\Delta t}{2\overline{c}_{\mathrm{c}}\,(jc)}\,k_5\,(jc)\,\varphi_1\,(jc)+\frac{2\Delta t Q_{\mathrm{c}}\,(jc)}{\overline{c}_{\mathrm{c}}\,(jc)}\right\}\;/\;\left\{\overline{\rho}_{\mathrm{c}}\,(jc)\right.\\ \left.+\frac{\Delta t w_2}{\Delta z\,(jc)\,A_{\mathrm{c}}\,(jc)}+\frac{k_5\,(jc)\,\Delta t h_{\mathrm{b2}}\,(jc)}{2\overline{c}_{\mathrm{c}}\,(jc)}\right\}\end{aligned}\tag{3.5-78}$$

with

$$h_{\mathrm{b1}}\,(jc)=\frac{1}{R_{\mathrm{ec1}}\,(jc)}+\frac{\gamma_2\,(jc)}{R_{\mathrm{sc1}}\,(jc)}\tag{3.5-79}$$

$$h_{\mathrm{b2}}\,(jc)=\frac{1}{R_{\mathrm{ec2}}\,(jc)}+\frac{\gamma_2\,(jc)}{R_{\mathrm{sc2}}\,(jc)}\tag{3.5-80}$$

and

$$\varphi_1\,(jc)=\frac{1}{2}\left\{\frac{T_{\mathrm{e2}}\,(jc)}{R_{\mathrm{ec2}}\,(jc)}+\frac{T_{\mathrm{e1}}\,(jc)}{R_{\mathrm{ec1}}\,(jc)}+\gamma_2\,(jc)\left[\frac{T_{\mathrm{si2}}\,(jc)}{R_{\mathrm{sc2}}\,(jc)}+\frac{T_{\mathrm{si1}}\,(jc)}{R_{\mathrm{sc1}}\,(jc)}\right]\right\}\tag{3.5-81}$$

If the inlet flow is positive, then the coolant temperature at node 1 is determined by the inlet temperature. Eq. (3.5.5.1) is then used to march up the channel through the inlet liquid slug, with $T_{\mathrm{c2}}\,(jc+1)$ being computed after $T_{\mathrm{c2}}\,(jc)$. Similarly, if the flow in the upper liquid slug is downward, then the assembly outlet reentry

temperature determines the coolant temperature at the last coolant node. Then an equation similar to Eq. (3.5-79) is used to march down through the upper liquid slug, with $T_{c2}(jc)$ being computed after $T_{c2}(jc+1)$.

The Eulerian calculations always go from node to node in the direction of flow. An inlet slug expelling downward and an outlet liquid slug going upward are special cases, since in these cases the calculation starts at a liquid vapor interface rather than an end of the subassembly. The interface liquid temperatures are first calculated using the Lagrangian treatment described below. Then Eq. (3.5.5.1) or the equivalent equation for downward flow is used to calculate the temperatures at the fixed nodes within the liquid slug. For the first fixed node near the interface, some of the terms in Eq. (3.5.5.1) are modified. The moving interface node is treated as node jc. The interface temperature is used for $T_{c2}(jc)$, and an interpolated value is used for $T_{c1}(jc)$. The distance from the fixed node to the interface at the end of the step is used for $\Delta z(jc)$. Interpolated interface cladding and structure temperatures are used in calculating ϕ_1 for the interface node.

3.5.5.2 Lagrangian Calculations for Interface Temperatures

For every liquid-vapor interface a vapor temperature is calculated at or very near the interface. The liquid temperature right at the interface would be close to the vapor temperature, but there can be strong axial temperature gradients in the liquid near the interface. These strong axial gradients would only extend a short distance into the liquid. The heat flow through the interface into a small vapor bubble is accounted for, as described in Chapter 12; but since only one liquid temperature node is used near the interface, the axial temperature distribution neat the interface is not represented. A liquid temperature is calculated for each interface, but axial conduction is neglected in this calculation. Thus, the liquid interface temperature can be considered as either the interface temperature that would occur if there were no axial conduction or the temperature a short distance from the interface where axial conduction is negligible.

A Lagrangian formulation, moving with the liquid, is used for the interface temperature calculation. The basic equation used is

$$\rho c \frac{\mathrm{DT_c}}{\mathrm{Dt}} = Q_c + Q_{ec} + Q_{sc} \tag{3.5-82}$$

where the Lagrangian total derivative is used. After finite differencing this equation gives

$$\begin{aligned}\rho_{ci} c_{ci} \frac{T_{li2}(k,L) - T_{li1}(k,L)}{\Delta t} &= Q_c(jc) + \frac{k_5(jc)}{2}\left[\frac{T_{ei1}(k,L) - T_{li1}(k,L)}{R_{eci1}(k,L)} + \frac{T_{ei2}(k,L) - T_{li2}(k,L)}{R_{sci1}(k,L)}\right] \\ &\quad + \gamma_2(jc)\left[\frac{T_{si1}(k,L) - T_{li1}(k,L)}{R_{sci1}(k,L)} + \frac{T_{si2}(k,L) - T_{li2}(k,L)}{R_{sci2}(k,L)}\right]\end{aligned} \tag{3.5-83}$$

or

$$T_{li2}(k,L) = \frac{T_{li1}(k,L)\left[1 - d_1\, h_{bi1}(k,L)\right] + d_1\left[\varphi_{1i}(k,L) + \frac{2Q_c(jc)}{k_5(jc)}\right]}{\left[1 + d_1 h_{bi2}(k,L)\right]} \tag{3.5-84}$$

where

$$d_1 = \frac{k_5(jc)\,\Delta t}{2\rho_{ci}\, c_{ci}} \tag{3.5-85}$$

$$h_{bi1}(k,L) = \frac{1}{R_{eci1}(k,L)} + \frac{\gamma_2(jc)}{R_{sci1}(k,L)} \tag{3.5-86}$$

$$h_{\text{bi2}}(k,L) = \frac{1}{R_{\text{eci2}}(k,L)} + \frac{\gamma_2(jc)}{R_{\text{sci2}}(k,L)} \tag{3.5-87}$$

k = bubble number

L = 1 for lower bubble interface, 2 for upper bubble interface

jc = coolant node containing the interface

$$\theta_{1\text{i}}(k,L) = \frac{T_{\text{ei2}}(k,L)}{R_{\text{eci2}}(k,L)} + \frac{T_{\text{ei1}}(k,L)}{R_{\text{eci1}}(k,L)} + \gamma_2(jc)\left[\frac{T_{\text{si2}}(k,L)}{R_{\text{sci2}}(k,L)} + \frac{T_{\text{si1}}(k,L)}{R_{\text{sci1}}(k,L)}\right] \tag{3.5-88}$$

T_{ei2}, T_{si2} = cladding and structure interface temperatures at the end of the step, extrapolated in time and interpolated between fixed cladding nodes.

T_{ei1}, T_{si1} = same at the beginning of the time step.

R_{eci2}, R_{sci2} = values of R_{ec} and R_{sc} at the interface at the end of the time step.

R_{eci1}, R_{sci1} = same at the beginning of the time step.

3.5.5.3 Lagrangian Calculation for Fixed Nodes

The Lagrangian temperature calculations for fixed coolant nodes are similar to those for interface nodes. The fluid particle that ends up at coolant node jc at the end of a time step is considered. During the time step, the particle travelled a distance

$$\Delta z' = \frac{(w_1 + w_2)\,\Delta t}{2\rho_{\text{c}}(jc)\,A_{\text{c}}(jj)} \tag{3.5-89}$$

where

$$jj = \begin{cases} jc & \text{if } w_1 + w_2 < 0 \\ jc - 1 & \text{otherwise} \end{cases} \tag{3.5-90}$$

At the beginning of the time step, the particle was at z', given by

$$z' = z_{\text{c}}(jc) - \Delta z' \tag{3.5-91}$$

The coolant temperature, T'_{c1}, at z' at the beginning of the step is obtained by linear interpolation between the nodes on either side of z'. Also, the cladding and structure temperatures, T'_{e1} and T'_{s1}, at z' at the beginning of the step are obtained by linear interpolation. The cladding and structure temperatures, T'_{e2} and T'_{s2}, at $z_{\text{c}}(jc)$ at the end of the time step are also obtained by linear interpolation between the cladding and structure nodes.

The result of finite differencing of Eq. (3.5-82) for the particle at node jc is

$$T_{\text{c2}}(jc) = \frac{\left\{T'_{\text{c1}}\left[1 - d'_1 h_{\text{b1}}(jj)\right] + d'_1\left[\theta'_1 + 2\frac{Q_{\text{c}}(jj)}{k_5(jj)}\right]\right\}}{\left[1 + d'_1\, h_{\text{b2}}(jj)\right]} \tag{3.5-92}$$

where

$$d'_1 = \frac{k_5(jj)\,\Delta t}{\rho_{\text{c}}(jc)\;\overline{c}_{\text{c}}(jj)} \tag{3.5-93}$$

and

$$\phi'_1 = \frac{T'_{\text{e2}}}{R_{\text{ec2}}(jj)} + \frac{T'_{\text{e1}}}{R_{\text{ec1}}(jj)} + \gamma_2(jj)\left[\frac{T'_{\text{s2}}}{R_{\text{sc2}}(jj)} + \frac{T'_{\text{s1}}}{R_{\text{sc1}}(jj)}\right] \tag{3.5-94}$$

Again, h_{b1} and $h_{\text{b2}}$are given by Eq. (3.5-79) and Eq. (3.5-80).

3.6 Fuel-Cladding Bond Gap Conductance

A number of gap-size-dependent bond gap correlations are available in SAS4A/SASSYS-1. The bond gap conductance depends on two main factors: the gap size or the contact pressure between fuel and cladding after the gap has closed, and the correlation for bond gap conductance as a function of gap size or contact pressure. Since small differences in differential expansion between fuel and cladding can make the difference between an open gap and a closed gap, and since gap conductance correlations are strongly dependent on gap size, the models used for fuel and cladding thermal expansion and swelling might have a much larger impact on computed bond gap conductances than the choice of the particular correlation used for bond gap conductance as function of gap size.

There are a number of options for computing the gap size. One common option for oxide fuel is to use DEFORM-IV to compute the steady-state and transient dimensions. Chapter 8 describes DEFORM-IV and the bond gap conductance correlations that can be used with it. A second option would be to use DEFORM-IV for the steady-state but not for the transient. In this case, the gap size and gap conductance determined in the steady-state calculations would be constant during the transient. A third option is not to use DEFORM-IV at all. In this case the gap size is constant, based on the user-specified pin dimensions, and the bond gap conductance is constant. The fourth option is to use a simple thermal expansion model for the transient bond gap size. For a metal fuel, the DEFORM-5 model described in Chapter 9 can be used to obtain the bond gap conductance.

The simple thermal expansion model applies only to the transient calculation. It can be used either with or without the DEFORM-IV steady-state calculations, but it cannot be used with the transient DEFORM-IV. In this model, it is assumed that the gap size, Δr_{g}, is determined by simple thermal expansion of the fuel and cladding from their steady-state dimensions:

$$\begin{aligned}\Delta r_{\mathrm{g}} &= r_{\mathrm{o}}\left(\mathrm{NE}\right) - r_{\mathrm{o}}\left(\mathrm{NR}\right) \\ &+ \frac{\left[r_{\mathrm{o}}\left(\mathrm{NE}\right) + r_{\mathrm{o}}\left(\mathrm{NE}'\right)\right]}{2}\alpha_{\mathrm{e}}\left[T\left(\mathrm{NE}\right) - T_{\mathrm{o}}\left(\mathrm{NE}\right)\right] \\ &- r_{\mathrm{o}}\left(\mathrm{NR}\right)\alpha_{\mathrm{f}}\left(\overline{T}_{\mathrm{f}} - \overline{T}_{\mathrm{fo}}\right)\end{aligned} \tag{3.6-1}$$

where

r_{o} = steady-state radii,

$\overline{T}_{\mathrm{o}}$ = steady-state temperature,

$\overline{T}_{\mathrm{f}}$ = average fuel temperature, mass-weighted average,

$\overline{T}_{\mathrm{fo}}$ = average steady-state fuel temperature,

α_{e} = cladding thermal expansion coefficient, and

α_{f} = fuel thermal expansion coefficient

The bond gap conductance then has the form

$$h_{\mathrm{b}} = \frac{\overline{h}_{\mathrm{b}}}{\Delta r_{\mathrm{g}}} \tag{3.6-2}$$

or

$$h_{\mathrm{b}} = A_{\mathrm{g}} + \frac{1}{B_{\mathrm{g}} + \frac{\Delta r_{\mathrm{g}} + C_{\mathrm{g}}}{\overline{h}_{\mathrm{b}}}} \tag{3.6-3}$$

depending on the correlation chosen. In these correlations $\overline{h}_{\mathrm{b}}$, A_{g}, B_{g}, and C_{g} are user-supplied correlation coefficients. For either correlation, the bond gap conductance is also constrained to lie between user-supplied minimum and maximum values; so if a value outside this range is calculated using Eq. (3.6-2) or Eq. (3.6-3), the minimum or the maximum value is used instead.

3.7 Fuel Pin Heat-transfer After Pin Disruption or Relocation of Fuel or Cladding

The preceding sections describe fuel pin heat transfer with intact fuel pins and no relocation of fuel of cladding. After pin disruption or the relocation of fuel or cladding, the heat transfer calculations are modified. The modifications after the start of in-pin fuel relocation in the PINACLE module are described in Chapter 15. The modification after the start of cladding melting and relocation in the CLAP module are described in Chapter 13. The modifications after pin disruption are described in Section 3.7.1 below and in Chapter 14 and Chapter 16.

3.7.1 Fuel-pin Heat Transfer After Pin Disruption in PLUTO2 or LEVITATE

When PLUTO2 or LEVITATE is active, the PLHTR subroutine calculates the heat conduction in all solid fuel (including axial blankets) and also in the cladding which is in contact with the lower and upper coolant slugs. The heat conduction calculation of the cladding in the interaction region, which is between the lower and upper coolant slug, (see Fig. 14.1-4) is performed in the PLUTO2 or LEVITATE modules (see Section 14.5.2 and Section 16.5.7) using a shorter time step than the PLHTR calculation.

Along the interaction region, the heat flow rate from the cladding inner surface to the fuel outer surface is calculated in PLUTO2 or LEVITATE assuming a constant gap conductance of the value in existence at the time of initiation. The PLUTO2 or LEVITATE calculated heat flow rates are integrated over a PLHTR time step in order to provide PLHTR with the total heat added during a heat-transfer time step. Outside the interacting region the heat flow rate between the liquid sodium flow and the cladding outer surface is calculated in the PLCOOL subroutine of PLUTO2. The latter subroutine mainly determines the liquid sodium temperatures in the coolant slugs. It uses the same time step as the PLHTR subroutine.

The temperature calculations in the molten fuel cavity in the pins are performed by PLUTO2 or LEVITATE and are part of the in-pin fuel motion calculation in these modules. The heat flow rates from each molten cavity node to the surrounding sold fuel are also calculated in PLUTO2 or LEVITATE. Since the time steps of the latter modules are shorter than the PLHTR time steps, the PLUTO2 or LEVITATE heat flow rates have to be integrated over the whole heat-transfer time step, because the total heat transferred to the cavity wall during a heat-transfer time step is required by PLHTR.

The initial configuration of the molten pin cavity at the time of pin failure is determined in the PLUTO2 and LEVITATE initialization routines PLINPT and PLSET (see Section 14.2.2). PLINPT initializes the integer array IXJ(K) for each axial node K with the index of the innermost radial fuel node whose melt fraction has not yet exceeded the input value FNMELT. This array IZJ(K) thus determines the initial molten cavity configuration.

When PLUTO2 or LEDVITATE are active, additional fuel can melt into the cavity and thereby enlarge it. The integer array IZJ(K) is updated for each axial node K whenever another radial node exceeds the input value FNMELT. However, such a radial node is only gradually added to the molten cavity (see Eq. (14.2-10) to Eq. (14.2-14)). The heat conduction calculation in PLHTR includes this partial node.

Figure 3.7.1 shows the radial grid used in PLHTR. The heat conduction calculation covers the radial region from I = INDBOT to I = NTHELP. The latter can be the outermost radial fuel node (for axial nodes in the interaction region) or the outer cladding node (for axial nodes outside the interaction region). Temperatures and heat sources are defined at the midpoints of the grid in Figure 3.7.1.

PLHTR is a modified version of the TSHTRV subroutine that calculates the fuel-pin heat transfer during coolant boiling and the reader is referred to Section 3.5 for a detailed presentation of the equations. One of the main differences is that the conduction calculation is done only in the solid fuel region and in the cladding outside the interaction region. This is achieved by having the calculational loops go from I = INDBOT to I = NTHELP (see Fig. 14.2-1) and by adding or subtracting the integrated heat flux to or from the solid fuel nodes at the

boundaries in the form of heat sources or sinks, respectively. The integrated heat flux at the outer pin boundary is obtained from

the array HFPICL, calculated in the PLUTO routine PLMISE or in the LEVITATE routine LESDEN. The integrated heat flux at the cavity boundary is obtained from the array HFCAWA; calculated in the PLUTO routine PC1PIN or the LVITATE routine LE1PIN. Moreover, the heat conduction terms at the fuel surface, which are necessary in the TSHTRV calculation, had to be set to zero. This meant setting the term BETA (NTHELP) in Eq. (3.3-49) to zero and ignoring all equations related to the cladding in Eq. (3.3-49).

3.8 Heat-transfer Time Step Control

Each channel uses its own separate heat-transfer time step size, so that channels in which temperatures are changing rapidly or in which boiling is occurring can use small heat-transfer time steps while other channels use larger steps.

After each heat-transfer time step, the size of the next step is determined, based on number of criteria. Most of these criteria are based on user-supplied values for maximum time step sizes and maximum temperature changes per step. The time step used in the smallest of the various criteria. If not other factor is more limiting, then the next heat-transfer step size is set equal to the initial main time-step size. This is usually in the range of .05-1.0 s. After the onset of boiling in a channel, a maximum boiling heat-transfer time step size, typically .01-.02 s, is used. In addition, a heat-transfer time step cannot go past the end of a main time step. Also, if the channel has not started boiling yet, then an attempt is made to end a heat-transfer time step right at, or very close to, the time when the first bubble is formed. For this purpose, the pre-boiling coolant routines make an estimate of the boiling time at the end of each coolant time step. The boiling time estimate is based on linear extrapolations in time for the coolant temperature and the saturation temperature at each axial node. The other criteria are based on the rate of change of the temperatures. The user supplies values of the maximum change per time step for the fuel and clad temperatures. Typical values are in the range of 30-50K. The rate of change of the fuel center-line temperature, the fuel surface temperature, and the clad mid-point temperature at each axial node are used to determine maximum time step sizes. After the minimum of the various time-step criteria has been found, the time step size is rounded to eight decimal places to minimize differences in results caused by different round-off errors on different computers. These criteria are summarized below:

1. Initial time step size
2. Maximum boiling time step size, if boiling
3. End of a main time step
4. Initiation of boiling
5. Maximum fuel center-line temperature change
6. Maximum fuel surface temperature change
7. Maximum clad mid-point temperature change

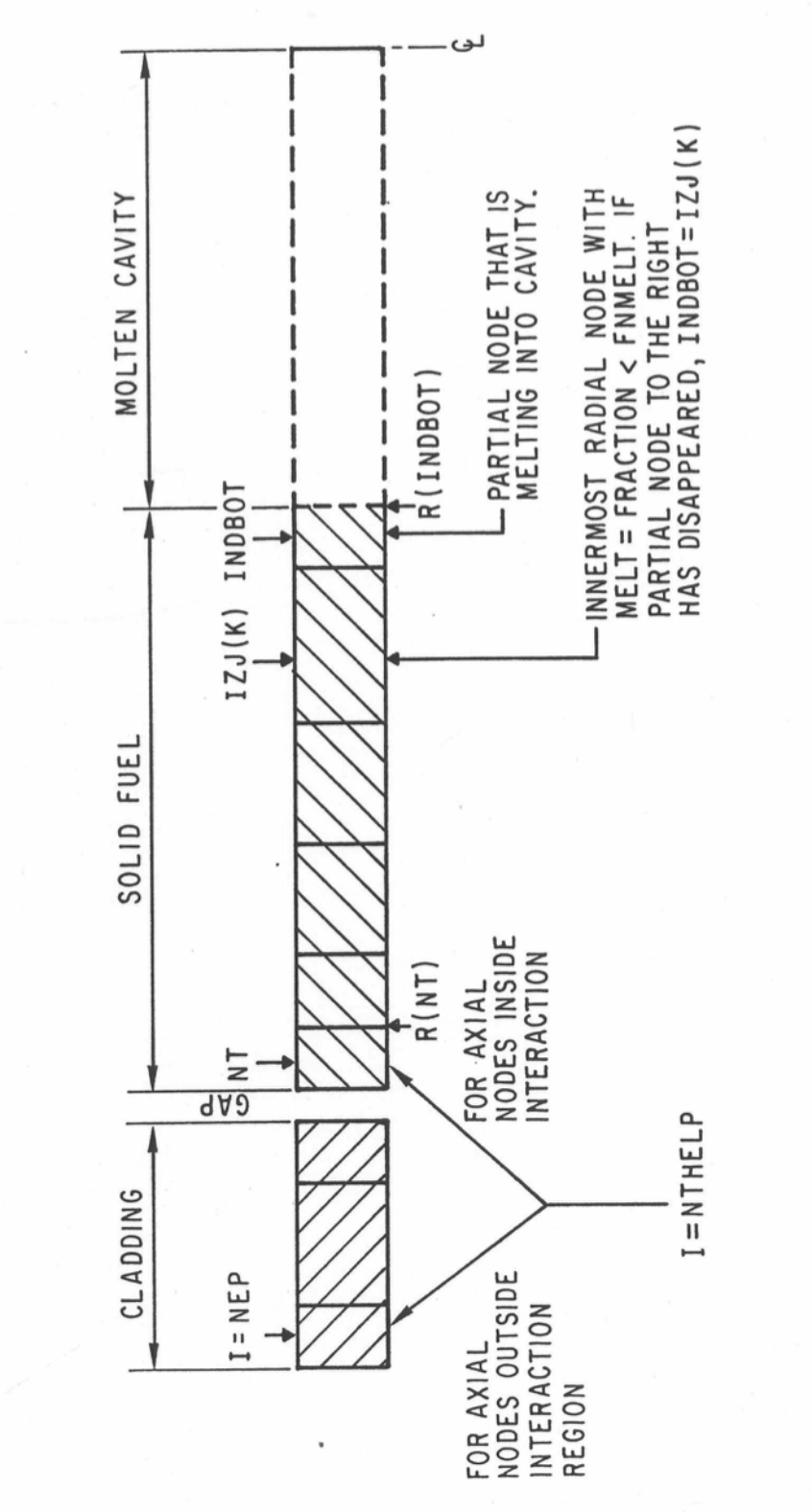


Figure 3.7.1: Radial Grid for the PLHTR Calculation

3.9 Steady-State and Single-Phase Transient Hydraulics

3.9.1 Introduction

The core assembly hydraulics treatment in SAS4A/SASSYS-1 includes the calculation of coolant flow rates and pressure distributions within each core channel. Coolant flow rates and pressures are calculated in a number of different places in the code. They are used in the steady-state thermal hydraulics initialization; the pre-voiding thermal hydraulics module calculates coolant flows and pressures; and after the onset of voiding, they are calculated in the boiling module. This section describes only the steady-state and pre-voiding calculations. Chapter 12 describes the hydraulics calculations after the onset of voiding. Chapter 14 and Chapter 16 describe the hydraulics calculations after pin disruption.

The coolant flow provides the heat removal from the fuel pins, and so the main reason for calculating coolant flow rates before the onset of voiding is to provide information for the fuel-pin temperature calculations. Core channel flow rates are also needed for the PRIMAR-4 primary loop thermal hydraulics calculations. The pressure distribution within a channel is calculated mainly to obtain the pressure-dependent coolant saturation temperature used to determine the onset of boiling.

For the steady-state initialization the user specifies the initial coolant flow rate for each channel and the outlet plenum pressure. The code then calculates the pressure distribution in each channel, starting from the outlet and working down to the inlet. The channel with the largest steady-state pressure drop is used to determine the steady-state inlet plenum pressure; and the inlet orifice coefficients in all other channels are adjusted so that all channels have the same total steady-state pressure drop.

In the transient hydraulic calculations, both the coolant flow rates and the pressure distributions are calculated for each channel. Pressure boundary conditions are used for the transient coolant flow rate calculations. The driving pressures for these calculations are the inlet and outlet coolant plenum pressures supplied by the PRIMAR-4 module. Since all channels use the same inlet and outlet pressures, flow redistribution between channels is automatically accounted for as temperatures and flows change. Extrapolated coolant temperatures are used to evaluate coolant properties in the transient hydraulics calculations, since coolant flow rates for a time step are calculated before temperatures are calculated.

Figure 3.1.2 shows the logic flow in subroutine TSCL0, the driver for pre-voiding core channel thermal hydraulics. As indicated previously, a multi-level time step approach is used for the transient calculations in SAS4A/SASSYS-1. In general, the routines in a transient module start with all quantities knows at the beginning of a time step, and a pass through a module results in calculating the values of relevant parameters of the module for one channel at the end of the time step. Different time-step sizes can be used for different phenomena. In particular, the pre-voiding coolant hydraulics time step can be smaller than, but not larger than, the heat-transfer time step or the PRIMAR time step. One pass through TSCL0 calculates one coolant time step for one channel. If the coolant time step finishes a heat-transfer step, then the temperature calculations for the heat-transfer step are also carried out.

Figure 3.9.1 shows the logic flow in subroutine TSCNV1. This subroutine and the routines called by it carry out the pre-viding coolant flow and pressure calculations. Subroutine TSCNV1 also checks on whether to switch to the boiling model.

As indicated in Figure 3.9.2, the pre-voiding transient hydraulics module interacts with PRIMAR and TSHTRN. PRIMAR supplies the inlet and outlet pressures that drive the coolant flows in the core channels. In return, TSCL0 supplies PRIMAR with the current channel flow rates, as well as hydraulic parameters to use in estimating future flow rates for the next PRIMAR time step. TSCL0 supplies the coolant flow rates to TSHTRN, and TSHTRN computes the coolant temperatures used by TSCL0.

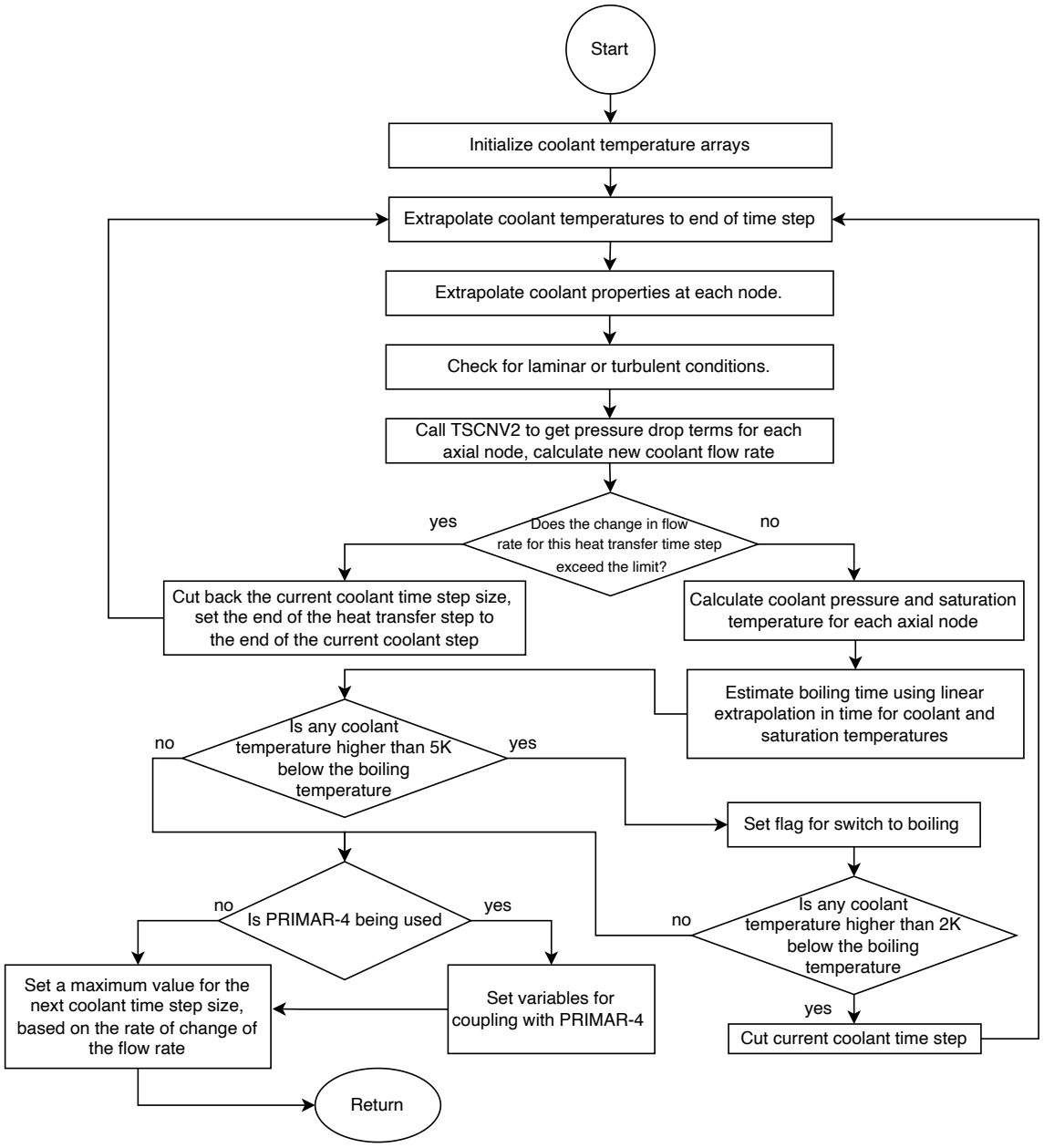


Figure 3.9.1: Subroutine TSCNV1, Pre-Boiling Coolant Flow Rates and Pressure Distribution

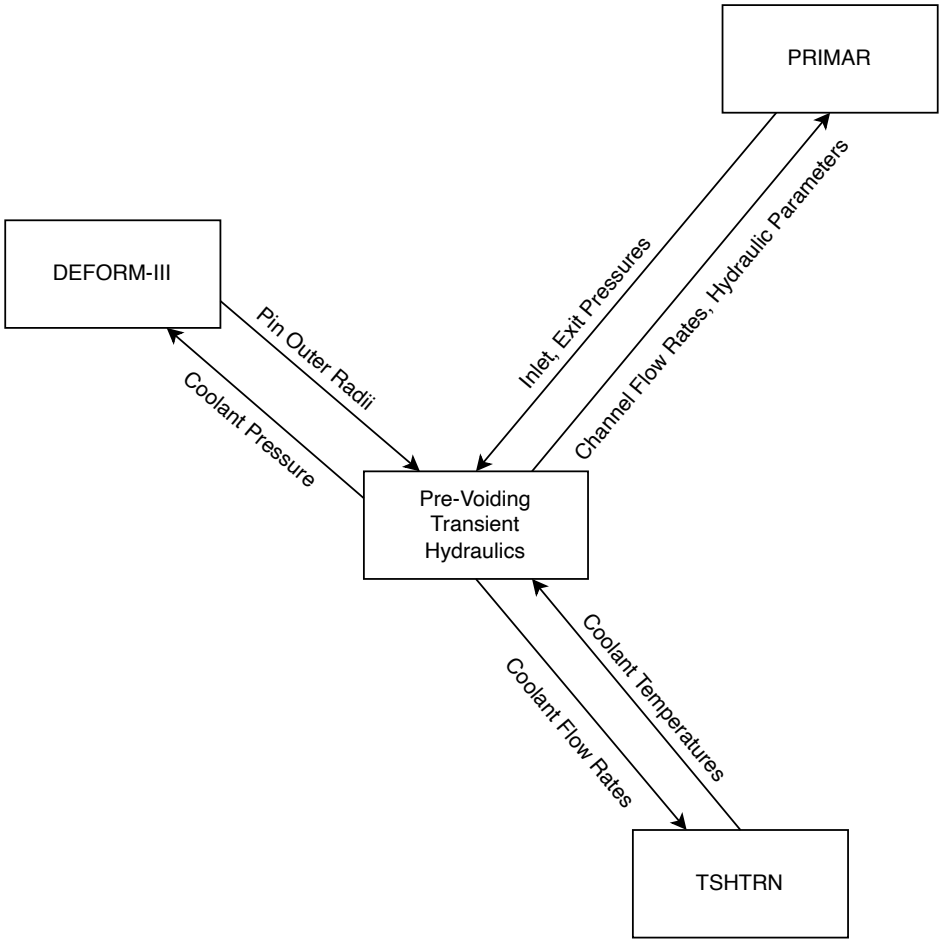


Figure 3.9.2: Interactions Between Pre-Voiding Transient Hydraulics and Other Modules

3.9.2 Basic Equations

Before the onset of voiding, the coolant is treated as incompressible, and the basic equation used for liquid coolant flow in non-voided channels is

$$\frac{1}{A_c}\frac{\partial w}{\partial t}+\frac{\partial p}{\partial z}+\frac{1}{A_c}\frac{\partial (wv)}{\partial z}=-\left(\frac{\partial p}{\partial z}\right)_{fr}-\left(\frac{\partial p}{\partial z}\right)_{K}-\rho_c g \tag{3.9-1}$$

with

$w = w(t)$ - independent of z

w = coolant flow rate (kg/s) $= \rho_c v A_c$

A_c = coolant flow area

p = pressure

ρ_c = density

v = coolant velocity (m/s)

z = axial position

t = time

$$\left(\frac{\partial p}{\partial z}\right)_{fr} = \text{friction pressure drop} \tag{3.9-2}$$

$$\left(\frac{\partial p}{\partial z}\right)_{fr} = f\frac{w\,|w|}{2\rho_c A_c^2 D_h} \tag{3.9-3}$$

f = friction factor, which is approximated differently depending on the chosen model option. When **IFRFAC** = 0:

$$f=\begin{cases} A_{fr}\,(Re)^{b_{fr}} & \text{if } Re \geq Re_L \\ \frac{A_{fL}}{Re} & \text{if } Re < Re_L \end{cases} \tag{3.9-4}$$

When **IFRFAC** = 1:

$$f = A_{fr}\,(Re)^{b_{fr}} + \frac{A_{fL}}{Re} \tag{3.9-5}$$

When **IFRFAC** = 2:

$$f=\begin{cases} A_{fr}\,(Re)^{b_{fr}} & \text{if } Re > Re_{TR} \\ \frac{A_{fL}}{Re}(1-\psi)^{C_1}(1-\psi^{C_2}) + A_{fr}\,(Re)^{b_{fr}}\,\psi^{C_3} & \text{if } Re_L \leq Re \leq Re_{TR} \\ \frac{A_{fL}}{Re} & \text{if } Re < Re_L \end{cases} \tag{3.9-6}$$

$$\psi = \frac{\log_{10}(\frac{Re}{Re_L})}{\log_{10}(\frac{Re_{TR}}{Re_L})} \tag{3.9-7}$$

where A_{fr} (**AFR**), b_{fr} (**BFR**), C_1 (**C1TRAN**), C_2 (**C2TRAN**), C_3 (**C3TRAN**), and A_{fL} (**AFLAM**) are user-supplied correlation coefficients and the correlations are in the form given by [3-16].

$Re = \frac{D_h|w|}{\mu A_c}$ = Reynolds number

Re_L = Maximum Reynolds number for flow in the laminar regime (**RELAM**)

Re_{TR} = Minimum Reynolds number for flow in the turbulent regime (**RETRAN**)

μ = viscosity

D_{h} = hydraulic diameter

$\left(\frac{\partial \mathrm{p}}{\partial \mathrm{z}}\right)_{\mathrm{K}}$ = orifice pressure drop

g = acceleration of gravity

ρ = density

The axial node structure shown in Figure 3.2.3 is used for the coolant. Figure 3.9.3 shows which variables are defined at node boundaries and which are averages or integrals over the length of a node. With this mesh structure, integrating Eq. (3.9-1) over the length of the channel gives

$$I_1 \frac{\partial \mathrm{w}}{\partial \mathrm{t}} + p_{\mathrm{t}} - p_{\mathrm{b}} + w^2 I_2 + A'_{\mathrm{fL}} w I_{3\mathrm{L}} + A'_{\mathrm{fr}} w \left|w\right|^{1+b_{\mathrm{fr}}} I_{3\mathrm{T}} + w \left|w\right| I_4(w) + g I_5 = 0 \tag{3.9-8}$$

where

$$I_1 = \int \frac{\mathrm{dz}}{A_{\mathrm{c}}} = \left(\frac{\Delta z_{\mathrm{i}}}{A}\right)_{\mathrm{b}} + \left(\frac{\Delta z_{\mathrm{i}}}{A}\right)_{\mathrm{t}} + \sum_{\mathrm{JC}=1}^{\mathrm{MZC}-1} X_{\mathrm{I1}}\left(\mathrm{JC}\right) \tag{3.9-9}$$

and

$$X_{\mathrm{I1}}\left(\mathrm{JC}\right) = \frac{\Delta z\left(\mathrm{JC}\right)}{A_{\mathrm{c}}\left(\mathrm{JC}\right)} \tag{3.9-10}$$

$\left(\frac{\Delta z_{\mathrm{i}}}{A}\right)_{\mathrm{b}}$ and $\left(\frac{\Delta z_{\mathrm{i}}}{A}\right)_{\mathrm{t}}$ are effective inertial terms at the bottom and top of the subassembly.

$$I_2 = \sum_{\mathrm{JC}=1}^{\mathrm{MZC}-1} X_{\mathrm{I2}}\left(\mathrm{JC}\right) \tag{3.9-11}$$

$$X_{\mathrm{I2}}\left(\mathrm{JC}\right) = \frac{1}{A_{\mathrm{c}}\left(\mathrm{JC}\right)^2}\left[\frac{1}{\rho_{\mathrm{c}}\left(\mathrm{JC}+1\right)} - \frac{1}{\rho_{\mathrm{c}}\left(\mathrm{JC}\right)}\right] \tag{3.9-12}$$

$$I_{3\mathrm{L}} = \int \frac{\mu}{2\rho_{\mathrm{c}} A_{\mathrm{c}} D_{\mathrm{h}}^2}\mathrm{dz} = \sum_{\mathrm{JC}=1}^{\mathrm{MZC}-1} X_{\mathrm{I3L}}\left(\mathrm{JC}\right) \tag{3.9-13}$$

$$X_{\mathrm{I3L}}\left(\mathrm{JC}\right) = \frac{\Delta z(\mathrm{JC})\overline{\mu}\left(\mathrm{JC}\right)}{\left[\rho_{\mathrm{c}}\left(\mathrm{JC}\right) + \rho_{\mathrm{c}}\left(\mathrm{JC}+1\right)\right] A_{\mathrm{c}}\left(\mathrm{JC}\right) D_{\mathrm{h}}\left(\mathrm{JC}\right)^2} \tag{3.9-14}$$

$$I_{3\mathrm{T}} = \int \frac{1}{2\rho_{\mathrm{c}} A_{\mathrm{c}}^2 D_{\mathrm{h}}}\left[\frac{D_{\mathrm{h}}}{\mu A_{\mathrm{c}}}\right]^{b_{\mathrm{fr}}}\mathrm{dz} = \sum_{\mathrm{JC}=1}^{\mathrm{MZC}-1} X_{\mathrm{I3T}}\left(\mathrm{JC}\right) \tag{3.9-15}$$

$$X_{\mathrm{I3T}}\left(\mathrm{JC}\right) = \frac{\Delta z(\mathrm{JC})}{\left[\rho_{\mathrm{c}}\left(\mathrm{JC}\right) + \rho_{\mathrm{c}}\left(\mathrm{JC}+1\right)\right] A_{\mathrm{c}}\left(\mathrm{JC}\right)^2 D_{\mathrm{h}}\left(\mathrm{JC}\right)}\left[\frac{D_{\mathrm{h}}\left(\mathrm{JC}\right)}{\overline{\mu}\left(\mathrm{JC}\right) A_{\mathrm{c}}\left(\mathrm{JC}\right)}\right]^{b_{\mathrm{fr}}} \tag{3.9-16}$$

$$I_4(w) = \sum_{\mathrm{JC}=1}^{\mathrm{MZC}-1} X_{\mathrm{I4}}\left(\mathrm{JC}\right) + \sum_{\mathrm{i}=1}^{\mathrm{NZONE}+1} \frac{1}{2\rho_{\mathrm{c}}\left(JC_i\right) A_i^2} K_i\left(Re_i\right) \tag{3.9-17}$$

$$X_{\mathrm{I4}}\left(\mathrm{JC}\right) = \frac{1}{\left[\rho_{\mathrm{c}}\left(\mathrm{JC}\right) + \rho_{\mathrm{c}}\left(\mathrm{JC}+1\right)\right] A_{\mathrm{c}}\left(\mathrm{JC}\right)^2} K_{\mathrm{or}}\left(\mathrm{JC}\right) \tag{3.9-18}$$

$$I_5 = \int \rho_{\mathrm{c}}\mathrm{dz} = \sum_{\mathrm{JC}=1}^{\mathrm{MZC}-1} X_{\mathrm{I5}}\left(\mathrm{JC}\right) \tag{3.9-19}$$

$$X_{I5}\,(\mathrm{JC}) = .5\ \left[\rho_c\,(\mathrm{JC}) + \rho_c\,(\mathrm{JC}+1)\right]\Delta z\,(\mathrm{JC}) \tag{3.9-20}$$

p_b = pressure at bottom of channel, and p_t = pressure at top of channel. A'_{fL} and A'_{fr} are defined as:

$$\begin{aligned} A'_{\mathrm{fL}} = 0 \text{ and } A'_{\mathrm{fr}} = A_{\mathrm{fr}} \quad & \text{if } \mathrm{Re} > \mathrm{Re_{TR}} \\ A'_{\mathrm{fL}} = A_{\mathrm{fL}}(1-\psi)^{C_1}(1-\psi^{C_2}) \text{ and } A'_{\mathrm{fr}} = A_{\mathrm{fr}}\psi^{C_3} \quad & \text{if } \mathrm{Re_L} \le \mathrm{Re} \le \mathrm{Re_{TR}} \\ A'_{\mathrm{fL}} = A_{\mathrm{fL}} \text{ and } A'_{\mathrm{fr}} = 0 \quad & \text{if } \mathrm{Re} < \mathrm{Re_L} \end{aligned} \tag{3.9-21}$$

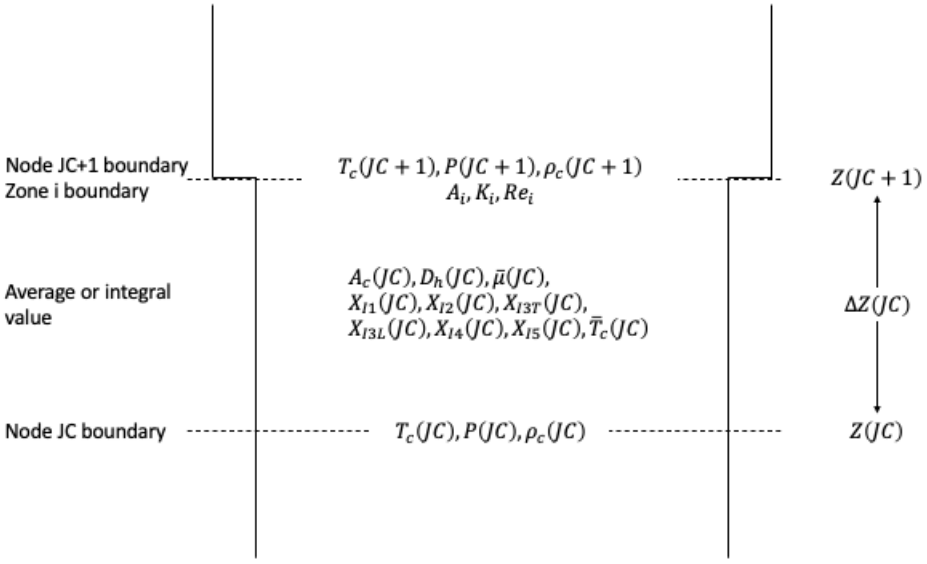


Figure 3.9.3: Definition of Variables on the Axial Coolant Mesh

3.9.3 Flow Orifices

As mentioned in the section on the SAS4A/SASSYS-1 channel treatment in Section 3.2, the channel is divided axially into a number of zones. One zone represents the fuel-pin region. Other zones represent regions above and below the fuel pins. At the bottom of each zone, flow direction-dependent orifice coefficients can be specified by the user. Additionally, orifice coefficients can be used at the top of the upper zone. These axial locations are hereafter called zone interfaces. The orifice coefficients at zone interfaces can represent orifice blocks, subassembly inlet and exit losses, and the pin support grid at the bottom of the pins. Users may define two different types of orifice coefficients at zone interfaces, a constant average coefficient: $K_{\mathrm{or}}\,(\mathrm{JC})$ and a Re-dependent interface coefficient, $K_i\,(Re_i)$. For each type of orifice, the user supplies one orifice coefficient for positive flow and another for negative flow (anisotropic values). The anisotropic constant average coefficients are defined using **XKORI**. The anisotropic Re-dependent coefficients are defined using **ZoneLossCoefTableID** and a Zone Interface Loss Coefficient Tables, described in Section 3.9.3.1.

In addition, in the fuel-pin region, constant average orifice coefficients representing equally-spaced grid spacers can be specified. These orifice coefficients are used as the values of $K_{\mathrm{or}}\,(\mathrm{JC})$ in Eq. (3.9-18) for the appropriate axial nodes. Grid spacer loss coefficients are defined using **NGRDSP** and **XKORGD**. Spacer grid loss coefficients are assumed to be the same in the forward and reverse directions.

The effect of each constant average orifice is distributed evenly over an axial node, unlike the Re-dependent interface coefficients which are concentrated at a point.

3.9.3.1 Anisotropic Re-dependent Loss Coefficients

Anisotropic Re-dependent loss coefficients can be defined at zone interfaces using Interface Loss Coefficient Tables. Zone interfaces are defined as the inlet of a channel axial zone and the outlet of a channel. By default, all zone interfaces are assumed to have a loss coefficient of zero. Only zone interfaces identified in an Interface Loss Coefficient Table will overwrite the default interface loss coefficient. The pressure drop at zone interface i caused by the loss coefficient is defined as

$$P_{i+} - P_{i-} = K_i(Re_i)\frac{w|w|}{2\rho_i A_i^2} \tag{3.9-22}$$

where

$$Re_i = \frac{Dh_i|w|}{A_i\mu_i} \tag{3.9-23}$$

$variable_i$ is the variable at interface i, and P_{i+}/P_{i-} are the pressure downstream and upstream the interface, respectively. In the core channels, the pressure reported at a channel zone interface is the upstream pressure, $P(JC_i) = P_{i-}$. At the channel outlet, the reported pressure is the downstream pressure, $P(MZC) = P_{(NZONE+1)+}$.

Two functional forms are currently available to capture the dependence of the loss coefficient on the Reynolds number:

$$K_i(Re_i) = \begin{cases} C_1 + C_2 Re_i^{C_3} & \text{Kind} = 1 \\ C_1 + C_2 exp\,(Re_i C_3) & \text{Kind} = 2 \end{cases} \tag{3.9-24}$$

$$\begin{aligned} C_1 &\geq 0 \\ C_2 &\geq 0 \\ C_3 &\geq -2 \end{aligned} \qquad \text{Kind} = 1$$

$$\begin{aligned} C_1 &\geq 0 \\ C_1 + C_2 &\geq 0 \end{aligned} \qquad \text{Kind} = 2 \tag{3.9-25}$$

Two sets of coefficients may be defined for a given zone interface; one set for forward flow and one set for reverse flow. Additionally, an interface area, A_i, and hydraulic diameter, Dh_i, may be defined for the evaluation of the interface pressure drop and interface Re. If an interface area and interface hydraulic diameter are not provided or are defined as zero, the area and hydraulic diameter of the node upstream or downstream the interface are used. If the upstream node has a smaller or equal area than the downstream zone, the upstream node's area and hydraulic diameter are used for the interface. If the upstream node has a bigger area than the downstream zone, the downstream node's area and hydraulic diameter are used for the interface.

At very low flow rates, the derivative of the loss coefficients may approach infinity. This introduces numerical difficulty when SAS attempts to reverse flow through a channel. In order to improve the robustness of the semi-implicit solution scheme, an optional lower limit has been introduced for Re-dependent loss coefficients, ReL. When the Reynold number drops below ReL, Eq. (3.9-24) is evaluated at ReL. Additionally, the derivative of $K_i(Re_i)$ is assumed to be zero.

Table 3.9.1: Input in a Core Channel Interface Loss Coefficient Table

Column Label	Description	Range
iZone	Index of the channel zone where the coefficient will be defined at the inlet. *iZone* = 1 is the inlet of the channel. *iZone* = *NZONE* + 1 is the outlet of the channel.	$1 \leq$ *iZone* $\leq$ *NZONE* + 1
Kind	Kind of functional form for calculating the anisotropic loss coefficient. See Eq. (3.9-24)	1 or 2
C1f, *C2f*, *C3f*	Function coefficients for forward flow	See Eq. (3.9-25)
C1b, *C2b*, *C3b*	Function coefficients for backward flow	See Eq. (3.9-25)
A (optional)	Area used to evaluate the Reynolds number and minor pressure loss at the interface. Optional area and hydraulic diameter must be provided as a pair.	$A \geq 0$
Dh (optional)	Hydraulic Diameter used to evaluate the Reynolds number at the interface. Optional area and hydraulic diameter must be provided as a pair.	$Dh \geq 0$
ReL (optional)	Lower Reynolds number limit for evaluating the functional form. For Reynolds numbers below the lower limit, the functional form is evaluated at *ReL* and the derivative is assumed to be 0.0.	$ReL \geq 0$

Example Channel Anisotropic Re-dependent Loss coefficient table input:

```
TABLE <id> CHAN1
iZone Kind  C1f    C2f       C3f   C1b    C2b     C3b   A            Dh   ReL
1     1     0.5 100000        -1   1.5      2       0  2.2          6.5   1.0
3     2     1       10 -0.000001     2 0.0001 0.00001 0.021        0.08   0.0
4     1     1   100000        -1   2.2  20000      -1 0.002 0.003183099  10.0
END
```

3.9.4 Finite Difference Equations - Coolant Flow Rates

If $w_1 = w(t)$, $w_2 = w(t + \Delta t)$, $\Delta w = w_2 - w_1$, then we approximate Δw as

$$\Delta w = \left[\theta_1 \left. \frac{\partial \mathrm{w}}{\partial \mathrm{t}} \right|_t + \theta_2 \left. \frac{\partial \mathrm{w}}{\partial \mathrm{t}} \right|_{t+\Delta t} \right] \Delta t \tag{3.9-26}$$

and

$$\theta_1 + \theta_2 = 1.0 \tag{3.9-27}$$

in which θ_2 is the degree of implicitness. For small time steps a semi-implicit calculation is used, with θ_1 = $\theta_2 = .5$. For a fully implicit calculation for large time steps, $\theta_1 = 0$ and $\theta_2 = 1.0$. The value used for θ_2 is

$$\theta_2 = \frac{6.12992 + 2.66054x + x^2}{2\ x\ 6.12992 + 3.56284x + x^2} \tag{3.9-28}$$

where

$$x = \Delta t / \tau \tag{3.9-29}$$

and τ is the time constant for flow rate changes given by

$$\tau = I_1/d_1 \tag{3.9-30}$$

The definition of d_1 is given by Eq. (3.9-37) below. This correlation for θ_2 is discussed in Section 3.19.1. The additional approximations are made that

$$(w_1 + \Delta w)^2 \simeq w_1^2 + 2\Delta w\, w_1 \tag{3.9-31}$$

$$(w_1 + \Delta w)\,|w_1 + \Delta w| \simeq w_1\,|w_1| + 2\,|w_1|\,\Delta w \tag{3.9-32}$$

$$(w_1 + \Delta w)\,|w_1 + \Delta w|^{1+b_{\rm fr}} \simeq w_1\,|w_1|^{1+b_{\rm fr}} + (2 + b_{\rm fr})\,|w_1|^{1+b_{\rm fr}}\ \Delta w \tag{3.9-33}$$

$$I_4(w_1 + \Delta w) \simeq I_4(w_1) + \frac{\partial I_4}{\partial Re}\frac{\partial Re}{\partial w}\Delta w \tag{3.9-34}$$

These equations are combined to give

$$\begin{aligned} I_1\frac{\Delta w}{\Delta t} &+ \theta_1\,(p_{\rm t1} - p_{\rm b1}) + \theta_2\,(p_{\rm t2} - p_{\rm b2}) + I_2\left(w_1^2 + 2\theta_2 w_1\Delta w\right) \\ &+ A'_{\rm fL}I_{\rm 3L}(w_1 + \theta_2\Delta w) + A'_{\rm fr}I_{\rm 3T}\left[w_1\,|w_1|^{1+b_{\rm fr}} + \theta_2\,(2 + b_{\rm fr})\,|w_1|^{1+b_{\rm fr}}\,\Delta w\right] \\ &+ (w_1|w_1| + \theta_2 2|w_1|\Delta w)\,I_4(w_1) + \theta_2 w_1|w_1|\frac{\partial I_4}{\partial Re}\frac{\partial Re}{\partial w}\Delta w + gI_5 = 0 \end{aligned} \tag{3.9-35}$$

where $p_{\rm t1}$ and $p_{\rm t2}$ are the pressures at the top of the subassembly at the beginning and end of the time step. Similarly, $p_{\rm b1}$ and $p_{\rm b2}$ are the pressures at the bottom of the subassembly at the beginning and end of the time step, respectively.

Solving for the coolant flow rate change, Δw, gives

$$\Delta w = \Delta t\frac{\left[\theta_1\,(p_{\rm b1} - p_{\rm t1}) + \theta_2\,(p_{\rm b2} - p_{\rm t2}) - I_2 w_1^2 - A'_{\rm fL}I_{\rm 3L}w_1 - A'_{\rm fr}I_{\rm 3T}w_1\,|w_1|^{1+b_{\rm fr}} - I_4(w_1)w_1\,|w_1| - gI_5\right]}{(I_1 + \Delta t\;\theta_2 d_1)} \tag{3.9-36}$$

where

$$d_1 = 2w_1I_2 + A'_{\rm fL}I_{\rm 3L} + (2 + b_{\rm fr})\,|w_1|^{1+b_{\rm fr}}A'_{\rm fr}I_{\rm 3T} + 2|w_1|I_4(w_1) + w_1|w_1|\frac{\partial I_4}{\partial Re}\frac{\partial Re}{\partial w} \tag{3.9-37}$$

The temperatures used to evaluate $\rho_{\rm c}$ and $\overline{\mu}$ in Eq. (3.9-12), Eq. (3.9-14), Eq. (3.9-16), Eq. (3.9-18) and Eq. (3.9-20) are extrapolated coolant temperatures, extrapolated to the end of the coolant step.

3.9.5 Coolant Pressures

After the flow rate has been calculated, the coolant pressures in a channel at the end of a time step are calculated. First, the pressure at node MZC, the last node at the top of the subassembly, is calculated as

$$p\,(\text{MZC}) = p_{\rm t2} + \left[\frac{\Delta z_{\rm i}}{A}\right]_{\rm t}\frac{\partial \rm w}{\partial \rm t} \tag{3.9-38}$$

where

$$p_{\rm t2} = p_{\rm x2} + \rho_{\rm cout}g\,[z_{\rm plu} - z_{\rm c}\,(\text{MZC})] \tag{3.9-39}$$

$z_{\rm plu}$ = outlet plenum elevation

ρ_{cout} = density at outlet

p_{x2} = coolant outlet plenum pressure at the end of the time step

Note that p_{x2}, the outlet plenum pressure supplied by the primary loop module, is defined at an elevation z_{plu}, the elevation of the upper plenum, whereas $p\,(\text{MZC})$ is calculated at $z\,(\text{MZC})$, so the gravity head term occurs in Eq. (3.9-39).

After $p\,(\text{MZC})$ has been calculated, the other pressures are calculated, starting at node MZC-1 working down, using

$$p\,(\text{JC}) = p\,(\text{JC}+1) + X_{\text{I1}}\,(\text{JC})\,\frac{\partial \text{w}}{\partial \text{t}} + w^2 X_{\text{I2}}\,(\text{JC}) + A'_{\text{fL}} w X_{\text{I3L}}\,(\text{JC}) + A'_{\text{fr}} w\,|w|^{1+b_{\text{fr}}}\,X_{\text{I3T}}\,(\text{JC}) + |w|\,w X_{I4}\,(\text{JC}) + g\,X_{I5}\,(\text{JC}) \quad (3.9\text{-}40)$$

and

$$p\,(\text{JC}) = \begin{cases} p\,(\text{JC}) + |w|\,w K_i / \left(2\rho_{\text{c}} A_i^2\right) & \text{JC} = \text{JC}_i \\ p\,(\text{JC}) & \text{JC} \neq \text{JC}_i \end{cases} \quad (3.9\text{-}41)$$

Note $\text{JC}_i = \text{MZC} - 1$ for the channel outlet, $i = \text{NZONE} + 1$.

In Eq. (3.9-38) and Eq. (3.9.5), the value used for $\frac{\partial w}{\partial t}$ is calculated at the end of the time step for the transient calculation as

$$\frac{\partial \text{w}}{\partial \text{t}} = \frac{\left[p_{\text{b2}} - p_{\text{t2}} - w_2^2 I_2 - A'_{\text{fL}} w_2 I_{\text{3L}} - A'_{\text{fr}} w_2\,|w_2|^{1+b_{\text{fr}}}\,I_{\text{3T}} - w_2\,|w_2|\,I_4(w_2) - g I_5\right]}{I_1} \quad (3.9\text{-}42)$$

The pressure, p_{b2} at the bottom of the subassembly at the end of the time step is calculated as

$$p_{\text{b2}} = p_{\text{in2}} - \rho_{\text{cin}} g\,[z_{\text{c}}\,(1) - z_{\text{pll}}] \quad (3.9\text{-}43)$$

z_{pll} = plenum location. Note that p_{in2}, the inlet plenum pressure supplied by the primary loop module, is defined at an elevation z_{pll}, the elevation of the lower plenum.

The pressure distribution in the steady state is calculated the same as in the transient, except that $\frac{\partial w}{\partial t}$ is zero in the steady-state calculation.

At the end of the steady state calculation, SAS will adjust the values of **XKORI** such that the pressure drop across each channel is consistent, given the user provided flow rates. These adjustments are reported in the standard output file. The magnitude of these adjustments can be limited using **ADJKTOL**. If **ADJKTOL** is set to a positive value and SAS determines an adjustment to **XKORI** that is larger than the defined tolerance is required, SAS will report an error and the simulation will be terminated.

3.10 Multiple-Pin Model

3.10.1 Introduction

The new multiple pin treatment was added to the code to account for pin-to-pin variations within a subassembly. This new multiple pin treatment can be used in at least two different ways. One approach is to use the new treatment to compute nominal or "best estimate" variations within a subassembly. Another approach is to compute hot channel behavior due to postulated deviations in coolant flow rate, coolant flow area, and pin power. When doing a whole-core analysis, one probably would not want the hot channel temperatures used in reactivity feedback calculation or in the core outlet temperatures that feed into the primary loop calculations.

Therefore, one would probably use parallel treatments for the same subassembly: a nominal one-pin or multi-pin treatment used for reactivity feedback, plus a hot channel treatment decoupled from the reactivity feedback and from the primary loop calculation. The hot channel treatment would be de-coupled by setting reactivity feedback coefficients to zero and by setting the number of subassemblies represented by this treatment to zero.

The new multiple pin treatment allows a number of coupled channels to be used to model a single subassembly. Thus a channel can represent a part of a subassembly instead of the whole subassembly. Peaking factors can be mechanistically calculated by reducing coolant flow areas and flow rates or increasing pin power levels in some channels.

As previously mentioned, the multiple pin option is currently only available for steady-state and single phase transient calculations.

3.10.2 Physical Model

In the new multiple pin option, the regions above and below the pin section of a subassembly are still represented by a single channel; but a number of channels can be used to represent the pin section. Each channel in the pin section can represent one or more concentric rows of pins and their associated coolant. It is also possible for channels to represent slices of pins for a subassembly with a strong lateral power skew or for one with a hot subassembly on one side and a cool subassembly on the opposite side. Figure 3.10.1 illustrates one way that a number of channels can be used to model the pin section of a subassembly as concentric rings of pins and coolant subchannels. The thimble flow region in this figure is a feature of the EBR-II XX09 instrumented subassembly and is not found in ordinary subassemblies. This case concentrates on the coolant subchannels and splits the pins. It is also possible to shift the channel boundaries half a pin and use a pin-based representation with intact pins and split coolant subchannels. Currently up to 56 channels can be used to represent a subassembly.

The new option accounts for coolant-to-coolant heat transfer between adjacent channels, including the effects of both conduction and turbulent mixing. It also accounts for subassembly-to-subassembly heat transfer from the duct wall of a subassembly, through the interstitial sodium, to the duct wall of a neighboring subassembly. In addition, axial conduction in the coolant is accounted for.

Figure 3.10.2 illustrates the axial representation of the subassembly flow, with parallel flow paths through the pin section. Each channel used to represent the pin section of subassembly has its own separate time-dependent flow rate, and the flow rates in all channels in a subassembly are driven by common pressures at the inlet and outlet of the pin section. Thus, transient flow redistribution among channels is accounted for. The single flow rate in the regions above and below the pin section is the sum of the pin section flow rates, so the subassembly flow orifice sees the correct total flow rate. Even though the coolant-to-coolant heat transfer coefficient includes a term for turbulent mixing between coolant subchannels, one effect that the flow calculation does not account for is mass flow between channels in the pin section, although cross-flow at the ends of the pin section is allowed. Therefore, if recirculation loops occur within a subassembly at low flows, the model would calculate them; but the recirculation loops would go to the ends of the pin section.

In the new model the coolant in channel I can transfer heat directly to the coolant in channel $\mathrm{I}-1$ and channel $\mathrm{I}+1$. Using correlations of the same form as those used in THI3D code [3-6] and the HOTCHAN code [3-7], the channel-to-channel heat flow per pin per unit height from channel I to channel $\mathrm{I}+1$ is calculated as:

$$Q_{\mathrm{I,I+1}} = \left[U_1 \overline{k} + U_2 \overline{C}\,(w_1 + w_{\mathrm{I+1}})\right](T_{\mathrm{I}} - T_{\mathrm{I+1}}) \tag{3.10-1}$$

where

$\overline{k}$ = average thermal conductivity of the coolant

$\overline{C}$ = average specific heat

w_{i} = coolant mass flow per pin (kg/s) in channel I,

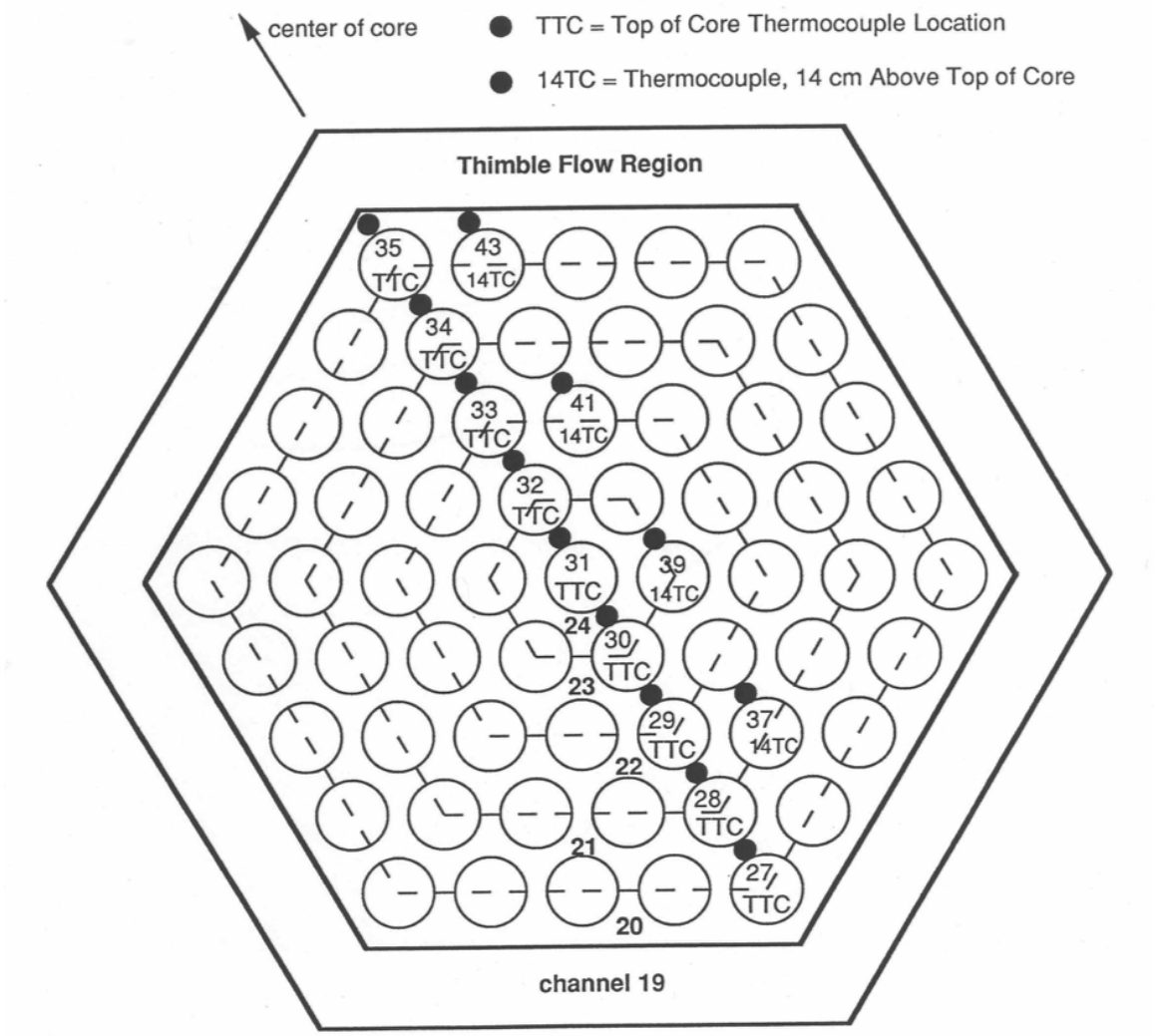


Figure 3.10.1: SAS4A/SASSYS-1 Multiple Pin Representation and Thermocouple Locations for the EBR-II XX09 Instrumented Subassembly

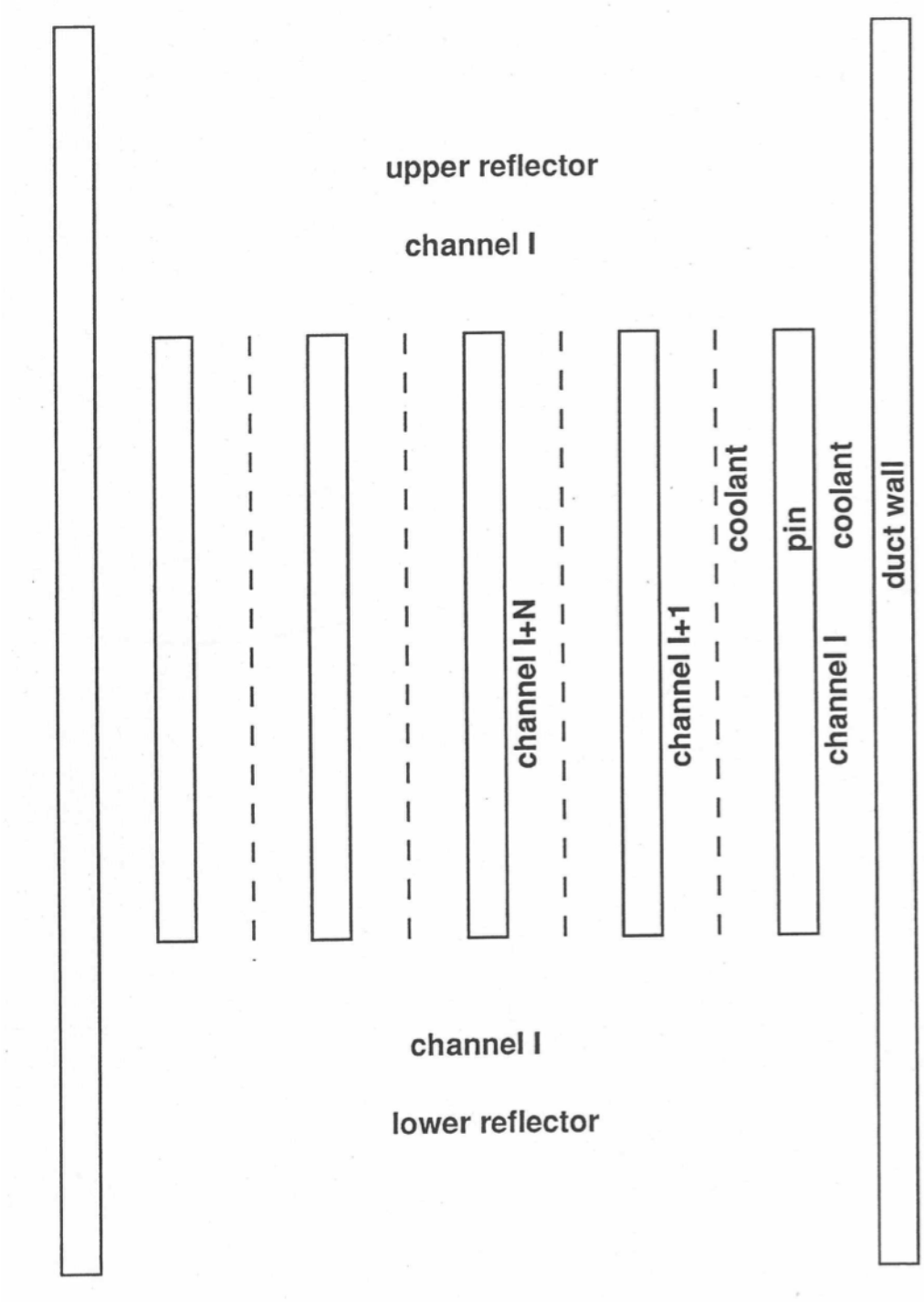


Figure 3.10.2: SAS4A/SASSYS-1 Multiple Pin Treatment of a Subassembly

and

T_I = coolant temperature

In this equation, U_I is a geometry factor for thermal conduction, and U_2 is a product of a turbulent-mixing coefficient and a geometry factor for turbulent mixing between channels. Since a SAS4A/SASSYS-1 channel usually models a group of coolant subchannels, the values used for U_1 and U_2 must account for a combination of parallel and series heat flow paths between the middle of channel I and the middle of channel $I+1$. Subassembly-to-subassembly heat transfer is handled in a somewhat simpler manner. For heat transfer from the outer surface of the structure in channel I to the outer surface of the structure in channel J, a constant value is used for the produce of the heat transfer coefficient and the heat transfer area per unit height.

3.10.3 Numerical Methods

Most of the transient heat transfer calculations and flow rate calculations in SASSYS-1 use semi-implicit time differencing in order to obtain stable and accurate solutions with reasonably long time steps. Before the onset of coolant boiling or pin disruption, time step sizes of a second or more are commonly used; and the code usually runs significantly faster than real time on a Cray XMP computer.

From a numerical computation point of view, the two main tasks in adding the multiple pin model to the code were the coolant-to-coolant heat transfer calculation and the coolant flow rate calculation with parallel flow paths in the pin section. Both of these calculations use semi-implicit time differencing. In the single-pin model, coolant temperatures for all of the radial temperature nodes in the pin, coolant, and structure at one axial node are solved for simultaneously in order to obtain a semi-implicit time differencing solution without iteration. In the new multiple-pin treatment, this concept is carried one step further. At a given axial node, temperature at all of the radial nodes for all channels representing a subassembly are solved for simultaneously. In the heat transfer calculations, the axial conduction terms, which are small, are treated with explicit forward time differencing so that axial nodes are decoupled and can be treated separately except for the coolant convection terms. The axial coupling due to the coolant convection terms is handled by starting at one end of the subassembly and solving for axial nodes one at a time in the direction of the flow. If flow has reversed in some channels but not in others, the calculation progresses in the direction of the dominate flow; and explicit forward differencing is used for the coolant convection terms in the non-dominate flow direction channels. The subassembly-to-subassembly heat fluxes are calculated with explicit forward differencing in time, and this does impose a stability limit on the time step size. For typical subassembly duct wall thicknesses, the explicit subassembly-to-subassembly heat flux calculation limits the maximum time step size to a value in the range from .25 to .5 seconds. For the coolant flow rate calculations, the incompressible flow momentum equations are linearized about conditions at the beginning of the time step. Then, flow at the end of the step are calculated for all channels in a subassembly simultaneously.

A null transient is used to obtain steady-state temperatures at the start of the regular transient. First all coolant, pin, structure, and reflector temperatures in all subassemblies are set to the coolant inlet temperature. Then, the power levels and coolant flow rates are held constant while a number of transient heat transfer time steps are made. Since the pin thermal time constant and the coolant transit time through a subassembly are both less than a second, the null transient results converge rapidly if reasonably large time steps are used.

3.10.4 Detailed Mathematical Treatment

The main computational parts of the new multiple pin model are the heat transfer calculations in the pin section of a subassembly, the coolant flow rate calculations for a subassembly, the subassembly-to-subassembly heat transfer, and changes to the driver routines to call the appropriate new routines at the proper times. The pin section heat transfer calculations are done in two new subroutines: TSHTM3 calculates temperatures in the core and axial blankets, and TSHTM2 calculates temperatures in the gas plenum region. The existing TSHTN1 still calculates temperatures in the reflector zones above and below the pin section, but it was modified. TSHTN1 had to be modified slightly to pick up mixed mean outlet temperatures from the multiple channels representing the pin section. Also, axial conduction in the coolant was added to TSHTN1. The new TSCLM1 routine calculates transient coolant flow rates for a subassembly. The subassembly-to-subassembly heat fluxes are calculated in CHCHFL.

3.10.4.1 Heat Transfer Calculations in the Core and Axial Blankets: Subroutine TSHTM3

The multiple pin heat transfer calculations for the core and axial blankets in subroutine TSHTM3 are similar to the single pin calculations in the existing subroutine TSHTN3, as described in Section 3.3.1. One difference is that TSHTN3 does one time step for one channel each time it is called, whereas TSHTM3 solves for temperatures in all pins or channels representing a subassembly when it is called. Also, TSHTM3 adds extra terms for coolant-to-coolant heat transfer; and TSHTM3 includes axial conduction in the coolant.

Within a fuel pin, one-dimensional radial heat transfer is used. The basic heat transfer equation is Eq. (3.3-1). This part of the calculation in TSHTM3 is identical to the corresponding calculation in TSHTN3.

For the coolant in channel I, the heat transfer calculations are basically one-dimensional in the axial direction, with extra terms for coolant-to-coolant heat transfer from channel $I-1$ and $I+1$. The basic heat transfer equation is:

$$\rho c A_{\mathrm{c}} \frac{\partial \mathrm{T}}{\partial \mathrm{t}} + \frac{\partial}{\partial \mathrm{z}} (wcT) = Q_{\mathrm{T}} A_{\mathrm{c}} \tag{3.10-2}$$

where

A_{c} = coolant flow area

z = axial position

w = coolant mass flow rate

and

Q_{r} = total heat source per unit volume

The heat source contains a number of terms:

$$Q_{\mathrm{T}} = Q_{\mathrm{c}} + Q_{\mathrm{ec}} + Q_{\mathrm{sc}} + Q_{\mathrm{ax}} + Q'_{\mathrm{I-1,I}} - Q'_{\mathrm{I,I+1}} \tag{3.10-3}$$

where

Q_{c} = source due to direct heating of the coolant by neutrons and gamma rays,

Q_{ec} = heat flow from cladding to coolant,

Q_{sc} = heat flow from structure to coolant,

Q_{ax} = axial conduction source, given by

$$Q_{\mathrm{ax}} = \frac{\partial}{\partial \mathrm{z}} k \frac{\partial \mathrm{T}}{\partial \mathrm{z}} \tag{3.10-4}$$

and

$$Q'_{\mathrm{I,I+1}} = \frac{Q_{\mathrm{I,I+1}}}{A_{\mathrm{c}}} \tag{3.10-5}$$

where $Q'_{\mathrm{I,I+1}}$ is given by equation 3.10-1. Note that Eq. (3.10-2) is the same as Eq. (3.3-5), except that additional source terms are included in the multiple pin model.

Finite differencing in space and time is used to solve these heat transfer equations. Figure 3.2.3 shows the axial and radial mesh used for a single channel. In the multiple pin model the pin section mesh is repeated for each channel, but the axial reflector zones are only used in the first channel used to represent a subassembly. At each axial node in the core and axial blankets a number of radial nodes are used for the fuel, three radial nodes are used for the cladding, one node is used in the coolant, and two nodes are used in the structure. All channels, representing a subassembly must use the same axial mesh. Also, all subassemblies connected by subassembly-to-subassembly heat transfer must use the same axial mesh.

For a given time step, Eq. (3.3-1) and Eq. (3.10-2) become linear finite difference equations. The temperatures are known at the beginning of the time step, and the temperature at the end of the step are the unknowns to be solved for. Semi-implicit time differencing is used for all terms except the axial conduction terms Q_{ax}, and the coolant-to-coolant terms $Q_{\mathrm{I-1,I}}$ and $Q'_{\mathrm{I,I+1}}$. The Q_{ax} axial conduction term is calculated explicitly based on temperatures at the beginning of the time step. The coolant-to-coolant terms are calculated fully implicitly, using the coolant flow rates and temperatures at the end of the step.

After finite differencing for one axial node, one obtains N simultaneous liner equations in N unknowns, where the unknowns are the temperatures at the end of the time step for all radial nodes in all channels representing the subassembly. These equations are solved by Gaussian elimination. The equations are basically tri-diagonal, with extra non-tri-diagonal terms for coolant-to-coolant heat transfer between channels. No full N by N matrix is ever set up by TSHTM3, and a general full matrix solution package is not used. Instead, only non-zero terms are computed and stored; and the Gaussian elimination solution was written specifically for this set of equations. The result is that the storage requirements vary linearly, rather than quadratically, with the maximum allowable value for N; and the computation time varies linearly, rather than quadratically, with the actual value of N.

3.10.4.2 Heat Transfer Calculations in the Gas Plenum Region: Subroutine TSHTM2

Subroutine TSHTM2 does the heat transfer calculations for one time step for all axial nodes in the gas plenum region for all channels representing a subassembly when it is called. TSHTM2 is similar to TSHTM3, but TSHTM2 deals with fewer radial nodes. As shown in Figure 3.2.5, in the plenum region there are still two radial nodes in the structure and one in the coolant; but there is only one in the cladding; and there are no fuel nodes. Instead of fuel temperatures there is one gas temperature common to all of the axial nodes in a pin. As in TSHTM3, for a given axial node TSHTM2 solves simultaneously for temperatures at all radial modes in all channels representing a subassembly.

One special problem in TSHTM2 is the gas temperature, which is common to all axial nodes in the gas plenum region of a pin. Axial nodes are handled one at a time, rather than simultaneously, whereas a semi-implicit time differencing treatment would require solving for all axial nodes simultaneously. The new multiple-pin TSHTM2 routine handles this problem in a simpler manner than the old single-pin TSHTN2. The way that the gas temperature is handled is TSHTM2 is to calculate a separate gas temperature for each axial node. At the beginning of the time step, the gas temperatures at all axial nodes are set to one common value. Then separate values are calculated for each axial node at the end of the step. Finally, an average of the separate values is calculated for the one common value at the ed of the step.

3.10.4.3 Coolant Flow Rates: Subassembly TSCLM1

The coolant for rate calculation for a subassembly in the multiple-pin routine TSCLM1 is much more complex than the corresponding calculation in the single pin routine TSCNV1. As shown in Figure 3.10.2, the multiple pin calculation involves multiple parallel flow paths in the pin section, in series with single flow paths in the reflector regions. Therefore, the multiple-pin routine TSCLM1 was written from scratch, rather than starting form the single pin TSCNV1 routine.

Figure 3.10.3 illustrates the main variables used in TSCLM1.

Incompressible flow is used for these calculations, so conservation of mass gives

$$N_1 \; w_{\mathrm{r}} = \sum_{\mathrm{k}} w_{\mathrm{pk}} N_{\mathrm{k}} \tag{3.10-6}$$

where

w_{r} = coolant mass flow rate per pin in the reflector zones,

w_{pk} = pin section coolant mass flow rate per pin in channel k,

N_{k} = number of pins in channel k,

and

I = channel number of the first channel representing the subassembly.

The momentum equation for the pin section is

$$\frac{L_{\mathrm{p}}}{A_{\mathrm{cpk}}} \frac{d\mathrm{w}_{\mathrm{pk}}}{\mathrm{dt}} = p_{\mathrm{b}} - p_{\mathrm{c}} - \Delta p_{\mathrm{pk}} \tag{3.10-7}$$

where

L_{p} = length of pins,

A_{cpk} = coolant flow area in channel k in the pin section,

t = time,

p_{b} = coolant pressure at the bottom of the pins,

p_{c} = coolant pressure at the top of the pins,

$$\Delta p_{\mathrm{pk}} = \text{ pressure loss } = \sum_{\mathrm{jv=JCPNBT}}^{\mathrm{JCPNTM}} \Delta p_{\mathrm{kjc}} \tag{3.10-8}$$

jc = axial coolant node,

JCPNBT = first axial node in the pin section

JCPNTM = last axial node in the pin section,

and

Δp_{kjc} = pressure loss in axial node jc of channel k.

The pressure loss contains a number of terms:

$$\Delta p_{\mathrm{kjc}} = \Delta p_{\mathrm{frkjc}} + \Delta p_{\mathrm{grkjc}} + \Delta p_{\mathrm{orkjc}} + \Delta p_{\mathrm{acckjc}} \tag{3.10-9}$$

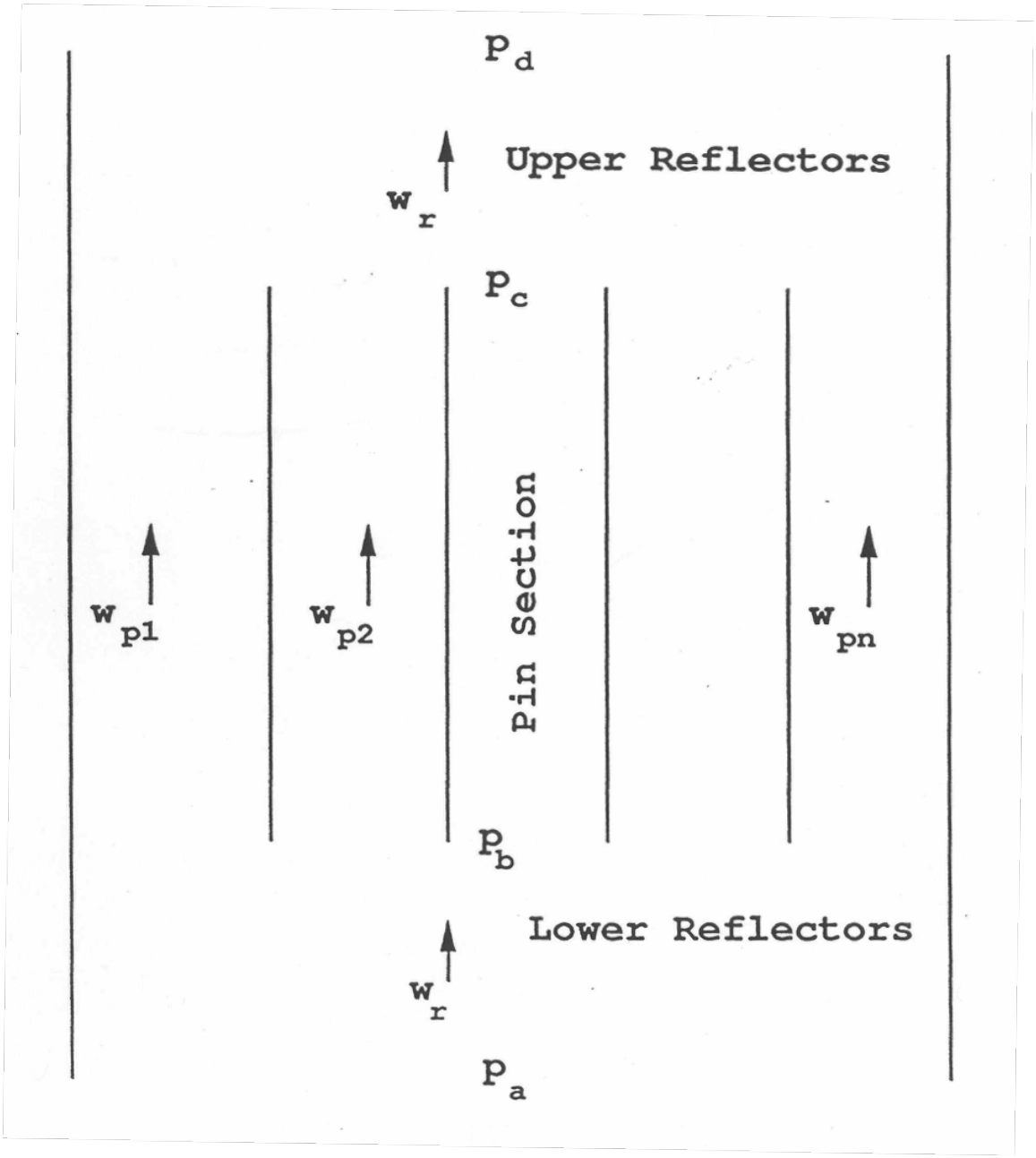


Figure 3.10.3: Subassembly Coolant Flows

where

$$\Delta p_{\text{frkjc}} = \text{ friction loss } = \frac{f_{\text{kjc}} \Delta z_{\text{jc}} w_{\text{pk}} \left| w_{\text{pk}} \right|}{2 \overline{\rho}_{\text{ckjc}} \, A_{\text{cpk}} \, D_{\text{hpk}}} \tag{3.10-10}$$

$$\Delta p_{\text{grpjc}} = \text{ gravity head } = g \overline{p}_{\text{ckjc}} \Delta z_{\text{jc}} \tag{3.10-11}$$

$$\Delta p_{\text{orfkjc}} = \text{ orifice loss } = \frac{K_{\text{orkjc}} w_{\text{pk}} \left| w_{\text{ph}} \right|}{2 \overline{\rho}_{\text{ckjc}} A_{\text{cpk}}^2} \tag{3.10-12}$$

$$\Delta p_{\text{acckjc}} = \text{ acceleration term } = \frac{w_{\text{pk}}^2}{A_{\text{cpm}}^2} \left(\frac{1}{\rho_{\text{ckjc+1}}} - \frac{1}{\rho_{\text{ckjc}}} \right) \tag{3.10-13}$$

Δz_{jc} = axial node length,

f_{kjc} = friction factor in channel k at node jc,

ρ_{ckjc} = coolant density at bottom of node jc in channel k,

$\overline{\rho}_{\text{ckjc}}$ = average coolant density in node jc in channel k,

D_{hpk} = Hydraulic diameter,

and

K_{orkjc} = orifice coefficient in node jc of channel k

The friction factor is calculated as in Eq. (3.9-4).

using

$$\text{Re} = \text{ Reynolds number } = \frac{D_{\text{hpk}} \left| w_{\text{pk}} \right|}{\mu_{\text{kjc}} \, A_{\text{cpk}}} \tag{3.10-14}$$

and

μ_{kjc} = viscosity of the coolant.

After semi-implicit finite differencing in time and linearization of the pressure drop terms, applying Eq. (3.10-7) to a time step from t to $t + \Delta t$ gives

$$\frac{L_{\text{p}}}{A_{\text{cpk}}} \frac{\Delta w_{\text{pk}}}{\Delta t} = p_{\text{b}}(t) + \theta_2 \Delta p_{\text{b}} - p_{\text{c}}(t) - \theta_2 \Delta p_{\text{c}} - \Delta p_{\text{pk}}(t) - \theta_2 \Delta w_{\text{pk}} K_{\text{wpk}} \tag{3.10-15}$$

where

$$\Delta w_{\text{pk}} = w_{\text{pk}}(t + \Delta t) - w_{\text{pk}}(t) \tag{3.10-16}$$

$$\Delta p_{\text{b}} = p_{\text{b}}(t + \Delta t) - p_{\text{b}}(t) \tag{3.10-17}$$

$$\Delta p_{\text{c}} = p_{\text{c}}(t + \Delta t) - p_{\text{c}}(t) \tag{3.10-18}$$

θ_2 = degree of implicitness

$$\theta_1 = 1 - \theta_2 \tag{3.10-19}$$

$$K_{\text{wpk}} = \sum_{\text{jc=JCPNBT}}^{\text{JCPNTM}} K_{\text{wpkjc}} \tag{3.10-20}$$

$$K_{\text{wpkjc}} = \frac{\partial \Delta \text{p}_{\text{kjc}}}{\partial \text{w}_{\text{pk}}} = K_{\text{frkjc}} + K_{\text{wcrkjc}} + K_{\text{acckjc}} \tag{3.10-21}$$

$$K_{\text{frkjc}} = \begin{cases} (2+b_{\text{fr}})\,\frac{\Delta p_{\text{frkjc}}(t)}{w_{\text{pk}}(t)} & \text{if } R_{\text{ckjc}} \geq R_{\text{et}} \\ \frac{\Delta p_{\text{frkjc}}(t)}{w_{\text{pk}}(t)} & \text{if } R_{\text{ckjc}} < R_{\text{et}} \end{cases} \tag{3.10-22}$$

or

$$K_{\text{frkjc}} = \left[(2+b_{\text{fr}})\,A_{\text{fr}} A\,(\text{RE})^{b_{\text{fr}}} + \frac{A_{\text{fL}}}{\text{RE}}\right] \frac{\Delta z_{\text{jc}}\,|w_{\text{pk}}(t)|}{2 p_{\text{ckjc}} D_{\text{hpk}} A_{\text{cpk}}} \tag{3.10-23}$$

$$K_{\text{wpkjc}} = \frac{2\Delta p_{\text{orfkjc}}(t)}{w_{\text{pk}}(t)} \tag{3.10-24}$$

and

$$k_{\text{acckjc}} = \frac{2\Delta p_{\text{acckjc}}(t)}{w_{\text{pk}}(t)} \tag{3.10-25}$$

Eq. (3.10-15) can then be written as

$$\Delta p_{\text{b}} - \Delta p_{\text{c}} = d_{0\text{pk}} + d_{0\text{pk}} + d_{1\text{pk}} \Delta w_{\text{pk}} \tag{3.10-26}$$

where

$$d_{0\text{pk}} = \frac{p_{\text{c}}(t) - p_{\text{b}}(t) + \Delta p_{\text{pk}}(t)}{\theta_2} \tag{3.10-27}$$

and

$$d_{1\text{pk}} = \frac{\frac{L_{\text{p}}}{A_{\text{cpk}}} + \theta_2 \Delta t K_{\text{wpk}}}{\theta_2 \Delta t} \tag{3.10-28}$$

Applying a similar process to the upper and lower reflector regions gives:

$$\Delta p_{\text{b}} = D_{0\text{l}} + D_{1\text{l}} \sum_{\text{k}} \Delta w_{\text{pk}} \tag{3.10-29}$$

and

$$\Delta p_{\text{c}} = D_{0\text{u}} + D_{1\text{u}} \sum_{\text{k}} \Delta w_{\text{pk}} \tag{3.10-30}$$

where

$$D_{0\text{l}} = \frac{\theta_1 p_{\text{a}}(t) + \theta_2 p_{\text{a}}(t+\Delta t) - p_{\text{b}}(t) - \Delta p_{\text{l}}(t)}{\theta_2} \tag{3.10-31}$$

$$D_{1\text{l}} = \frac{-\sum_{\text{KA}=1}^{\text{KZPIN}-1} \frac{L_{\text{kz}}}{A_{\text{cKZ}}} - \theta_2 \Delta t K_{w\text{l}}}{\theta_2 \Delta t} \tag{3.10-32}$$

$$D_{0\text{u}} = \frac{\theta_1 p_{\text{d}}(t) + \theta_2\, p_{\text{d}}(t+\Delta t) - p_{\text{c}}(t) + \Delta p_{\text{u}}(t)}{\theta_2} \tag{3.10-33}$$

$$D_{1\text{u}} = \frac{\sum_{\text{KZ}-\text{KZPIN}+1}^{\text{KZM}} \frac{\text{L}_{\text{KZ}}}{A_{\text{cKZ}}} + \theta_2 \Delta t K_{\text{wu}}}{\theta_2 \Delta t} \tag{3.10-34}$$

L_{KZ} = length of zone kz

A_{cKZ} = coolant flow area in zone kz

KZPIN = axial zone number of the pin section

KZM = last zone number

$$\Delta p_{\mathrm{l}} = \sum_{\mathrm{jc}=1}^{\mathrm{JCPNBT}-1} \Delta p_{\mathrm{rjc}} \tag{3.10-35}$$

$$\Delta p_{\mathrm{u}} = \sum_{\mathrm{jc}=\mathrm{JCPNTP}}^{\mathrm{MZCM1}} \Delta p_{\mathrm{rjc}} \tag{3.10-36}$$

Δp_{rjc} = pressure loss in node jc of a reflector zone

JCPNBT = first axial node in pin section

JCPNTP = first axial node above the pin section

MZCM1 = last axial node

$$K_{\mathrm{wl}} = \frac{\partial \Delta p_{\mathrm{l}}}{\partial w_{\mathrm{r}}} \tag{3.10-37}$$

and

$$K_{\mathrm{wu}} = \frac{\partial \Delta p_{\mathrm{u}}}{\partial w_{\mathrm{u}}} \tag{3.10-38}$$

Then, combining Eq. (3.10-26) and Eq. (3.10-29) gives

$$\Delta p_{\mathrm{b}} = B_{\mathrm{o}} + B_1 \Delta p_{\mathrm{c}} \tag{3.10-39}$$

where

$$B_{\mathrm{o}} = \frac{D_{\mathrm{ol}} - D_{1\mathrm{l}} S_1}{1 - D_{1\mathrm{f}} S_{\mathrm{o}}} \,, \tag{3.10-40}$$

$$B_1 = \frac{-D_{1\mathrm{l}} S_{\mathrm{o}}}{1 - D_{1\mathrm{l}} S_{\mathrm{o}}} \tag{3.10-41}$$

$$S_{\mathrm{o}} = \sum_{\mathrm{k}} \frac{1}{d_{1\mathrm{pk}}} \,, \tag{3.10-42}$$

and

$$S_1 = \sum_{\mathrm{k}} \frac{d_{\mathrm{opk}}}{d_{1\mathrm{pk}}} \,, \tag{3.10-43}$$

Similarly, combining Eq. (3.10-26) and Eq. (3.10-30) gives

$$\Delta p_{\mathrm{c}} = C_{\mathrm{o}} + C_1 \Delta p_{\mathrm{b}} \tag{3.10-44}$$

where

$$C_{\mathrm{o}} = \frac{D_{\mathrm{ou}} - D_{1\mathrm{u}} S_1}{1 + D_{1\mathrm{u}} S_{\mathrm{o}}} \,, \tag{3.10-45}$$

and

$$C_1 = \frac{D_{1\mathrm{u}} S_{\mathrm{o}}}{1 + D_{1\mathrm{u}} S_{\mathrm{o}}} \,. \tag{3.10-46}$$

Combining Eq. (3.10-39) and Eq. (3.10-44) gives

$$\Delta p_{\mathrm{c}} = \frac{C_{\mathrm{o}} + C_1 B_{\mathrm{o}}}{1 - C_1 B_1} \tag{3.10-47}$$

Then Δp_{b} can be obtained from Eq. (3.10-39) and Δw_{bk} can be obtained from Eq. (3.10-26) for each channel.

After p_{b}, p_{c}, and the coolant flow rates have been calculated for the end of the time step it is possible to calculate the pressures. The pressure is calculated at the axial node boundaries. p_{kjc} is the pressure at the bottom (inlet end) of node jc in channel k. First, the nodal pressure loss at $t + \Delta t$ is calculated as

$$\Delta p_{\mathrm{pkjc}}\left(t+\Delta t\right) = \Delta p_{\mathrm{kjc}}\left(t\right) + K_{\mathrm{wpkjc}} \Delta w_{\mathrm{pk}} \tag{3.10-48}$$

and $\Delta p_{\mathrm{pk}}(t + \Delta t)$ is calculated from Eq. (3.10-8). Then Eq. (3.10-7) is used to obtain $\frac{\mathrm{dw_{pk}}(t+\Delta t)}{\mathrm{dt}}$.

Integrating the momentum equation over one axial node gives

$$\frac{\Delta z_{\mathrm{j}}}{A_{\mathrm{cpk}}} \frac{\mathrm{dw_{pk}}}{\mathrm{dt}} = p_{\mathrm{pkjc}} - p_{\mathrm{pkjc+1}} - \Delta p_{\mathrm{kjc}} \tag{3.10-49}$$

Starting by setting

$$p_{\mathrm{pkjJCPNTP}} = P_{\mathrm{c}} \tag{3.10-50}$$

the code marches down the channel, using Eq. (3.10-49) to obtain p_{pkjc} after $p_{\mathrm{pkjc+1}}$ has been calculated. A similar procedure is used for calculating the pressures in the upper and lower reflector zones.

The equation used to compute the degree of implicitness as a function of time step size in TSCMV1 is

$$\theta_2 = \frac{a + bx + x^2}{2a + cx + x^2} \tag{3.10-51}$$

where

$$x = \Delta t / \tau \tag{3.10-52}$$

τ = a time constant

a = 6.12992

b = 2.66054

and

c = 3.56284

The basis for this expression is given in Section 3.19.1. For a single channel treatment, the time constant, τ would be

$$\tau = \frac{\sum_{\mathrm{KZ}=1}^{\mathrm{KZM}} \frac{L_{\mathrm{KZ}}}{A_{\mathrm{cKZ}}} + \left(\frac{L_{\mathrm{i}}}{A}\right)_{\mathrm{i}} + \left(\frac{L_{\mathrm{i}}}{A}\right)_{\mathrm{x}}}{K_{\mathrm{wpk}}\, K_{\mathrm{w1}} + K_{\mathrm{wu}}} \tag{3.10-53}$$

where

$\left(\frac{L_{\mathrm{i}}}{A}\right)_{\mathrm{i}}$ = extra coolant inertia term at the subassembly inlet to account for inertia of the coolant in the inlet plenum,

and

$\left(\frac{L_{\mathrm{i}}}{A}\right)_{\mathrm{x}}$ = same for the subassembly outlet.

Since a simultaneous solution of flows in all channels of an assembly plus the lower and upper reflector zones is required, the overall time constant is calculated as

$$\tau = \frac{\frac{L_{\mathrm{p}}}{\sum_{\mathrm{k}} A_{\mathrm{cpk}}} + \sum_{\mathrm{kZ}\neq\mathrm{KZPIN}} \frac{L_{\mathrm{KZ}}}{A_{\mathrm{cKZ}}} + \left(\frac{L_{\mathrm{i}}}{A}\right)_{\mathrm{i}} + \left(\frac{L_{\mathrm{i}}}{A}\right)_{\mathrm{x}}}{K_{\mathrm{w1}} + K_{\mathrm{wu}} + \frac{1}{\sum_{k} \frac{1}{K_{\mathrm{wpk}}}}} \tag{3.10-54}$$

3.10.5 Relationship Between Single Pin and Multiple Pin Models

The single pin treatment and the multiple pin treatment coexist in the SAS4A/SASSYS-1 code; it is possible to use single pin treatments for some subassemblies and multiple pin treatments for other subassemblies in the same problem. There are now two separate sets of subroutines in the code for computing single phase thermal hydraulics: the single pin routines and multiple pin routines. Since the multiple pin model can handle the case of a single pin per subassembly, the single pin single phase thermal hydraulics subroutines are now redundant; and at some point in the future they will probably be removed from the code.

As previously mentioned, currently the multiple pin model is only available for steady state and single phase transient calculations; no multiple pin treatment is available in the code for coolant boiling, pin disruption, or relocation of fuel or cladding. In the future, the multiple pin treatment will probably be extended to coolant boiling and to relocation of fuel and cladding.

3.11 Subassembly-to-subassembly Heat Transfer

SAS4A/SASSYS-1 contains a model for transferring heat from the duct wall of one subassembly, through the interstitial sodium, to the duct wall of an adjacent subassembly. During normal, full power operation the subassembly-to-subassembly heat flow is usually small compared to the power generation rate, but at decay heat power levels the heat flow between a subassembly and its neighbors can be comparable to the power generation rate within the subassembly. This heat flow between adjacent subassemblies will affect subassembly-to-subassembly flow re-distribution at low flow rates; and at low powers and flow rates it will tend to cause all subassemblies to have similar temperature rises.

The subassembly duct wall is normally represented by the structure in a SAS4A/SASSYS-1 channel; so this mode is implemented by transferring heat from the outer structure nodes of a SAS4A/SASSYS-1 channel to the outer structure nodes of other SAS4A/SASSYS-1 channels or to a constant temperature heat sink. Also, it is possible to transfer heat from the outer structure node of one channel to the coolant of another channel. This option was included in the code in order to treat the thimble flow region shown in Figure 3.10.1. The equation for the structure to structure heat flux, $Q_{\mathrm{I,M}}(j)$, from channel I to channel M at axial node j is

$$Q_{\mathrm{I,M}}(j) = \frac{(\mathrm{HA})_{\mathrm{I,M}}}{N_{\mathrm{PI}} N_{\mathrm{SI}}} \left[T_{\mathrm{sto}}(I, J) - T_{\mathrm{sto}}(M, j) \right] \tag{3.11-1}$$

where

$(\mathrm{HA})_{\mathrm{I,M}}$ = heat transfer coefficient times area per unit height, between channels I and M

T_{sto} = structure outer node temperature

N_{PI} = number of pins per subassembly in channel I

N_{SI} = number of subassemblies represented in channel I

The SAS4A/SASSYS-1 channel treatment only accounts for one structure per channel; so the channel-to-channel heat flux seen by the structure in a channel will be a sum of the heat fluxes to all adjacent channels. This heat flux is

$$Q_{\mathrm{chch,I}}(j) = \sum_{\mathrm{M}} Q_{\mathrm{I,M}}(j) \tag{3.11-2}$$

Each SAS4A/SASSYS-1 channel can transfer heat to up to eight other channels. A subassembly has six nearest neighbors, but a channel can represent more than one subassembly. Therefore, a channel can be in contact with more than six other channels.

If the constant temperature heat sink option is used, then the user supplies values of T_{snk} and H_{snk} for each axial zone in the channel, and these values are held constant during the calculations.

In contrast to many of the other temperature calculations in the code, explicit forward time differencing is used for the channel to channel heat transfer. The channel to channel heat fluxes are calculated at the beginning of each main time step using the temperatures at that time. These heat fluxes are then held constant during a time step. This explicit forward differencing imposes stability limits on the time step size. The stability limit is typically in the range of .2-.5 seconds.

At this point, the subassembly-to-subassembly heat transfer model has only been implemented in the transient single phase heat transfer calculations. It is necessary to run a null transient to obtain the current steady-state temperatures when subassembly-to-subassembly heat transfer is used.

3.12 Interaction with Other Models

As mentioned in Section 3.1, the heat-transfer routines interact with a number of other models. These interactions are indicated in Figure 3.1.1.

If PRIMAR-4 is being used, then the subassembly coolant outlet temperatures calculated in the heat-transfer routines are used by PRIMAR-4 to calculate the outlet plenum temperature. Also, if flow reversal occurs in a channel, then the temperature calculated in the heat-transfer routines for the coolant leaving the bottom of the subassembly is used by PRIMAR-4 in the calculation of the inlet plenum temperature. Section 3.3.6 describes how the inlet and outlet plenum temperatures are used in the calculations of the subassembly coolant inlet and reentry temperatures.

Before the onset of voiding, TSHTRN calculates the coolant temperatures used in the coolant channel hydraulics calculations, and the hydraulics routines calculate the coolant flow rates used by TSHTRN.

After the onset of voiding, coolant temperatures are calculate din TSBOIL. This module supplies the heat flux at the cladding outer surface, as defined in Eq. (3.5-2) and used in Eq. (3.5-9) to TSHTRV. TSBOIL uses the calculated cladding temperatures form TSHTRV to obtain extrapolated cladding temperatures, as in Eq. (3.5.1), for use in its coolant temperature calculations.

The point kinetics model supplies the power level used in the heat-transfer routines. In return, the heat-transfer routines supply the temperatures used to calculate reactivity feedback.

3.12.1 Reactivity Feedback

The temperatures calculated in the core thermal hydraulics routines are used to calculate various components of reactivity feedback. These reactivity components include Doppler feedback, axial expansion of the fuel and cladding, density changes in the sodium, core radial expansion and control rod drive expansion. These reactivity feedbacks are described in Chapter 4.

3.12.2 Coupling Between Core Channels and PRIMAR-4

As described in Section 5.2.2, the PRIMAR-4 calculations for a PRIMAR time step are carried out before the core channel coolant calculations. PRIMAR-4 must make estimates of the core flows for a new time step, and it also makes corrections for the differences between its estimates for the previous step and values computed by the core channel coolant routines. The coolant routines supply information to PRIMAR-4 for use in these estimates and corrections. Also, PRIMAR-4 supplies inlet and outlet coolant plenum pressures and temperatures for the use in the core channel calculations.

3.12.2.1 Supplied By PRIMAR-4

The information supplied by PRIMAR-4 is

$p_{\text{in}}(t_{\text{p1}})$ = inlet plenum pressure at the beginning of the PRIMAR time step

$p_{\text{x}}(t_{\text{p1}})$ = outlet plenum pressure at the beginning of the PRIMAR time step

$\frac{\text{dp}_{\text{in}}}{\text{dt}}$ =time derivative of the inlet plenum pressure

$\frac{\text{dp}_{\text{x}}}{\text{dt}}$ =time derivative of the outlet plenum pressure

ρ_{cin} =coolant density in the inlet plenum

ρ_{cout} =coolant density in the outlet plenum

T_{cin} =coolant temperature in the inlet plenum

and

T_{cout} = coolant temperature in the outlet plenum

The pressure p_{in} is at an elevation z_{pll} and p_{x} is at z_{plu}. At any time, t, during the PRIMAR time step, the inlet plenum pressure is

$$p_{\text{in}}(t) = p_{\text{in}}(t_{\text{p1}}) + (t - t_{\text{p1}})\frac{\text{dp}_{\text{in}}}{\text{dt}} \tag{3.12-1}$$

and the exit plenum pressure is

$$p_{\text{x}}(t) = p_{\text{x}}(t_{\text{p1}}) + (t - t_{\text{p1}})\frac{\text{dp}_{\text{x}}}{\text{dt}} \tag{3.12-2}$$

3.12.2.2 Supplied To PRIMAR-4

The information supplied to PRIMAR-4 by the core coolant routines for channel ic at the subassembly inlet $(L = 1)$ or outlet $(L = 2)$ is the following:

$$\Delta m_{\text{c}}(L) = \sum_{\text{ic}} N_{\text{ps}}(\text{ic}) \int_{t_{\text{p1}}}^{t_{\text{p2}}} w(L, ic)\text{dt}\ , \tag{3.12-3}$$

$$\Delta m_{\text{c}} T_{\text{c}}(L) = \sum_{\text{ic}}\ N_{\text{ps}}(\text{ic}) \int_{t_{\text{p1}}}^{t_{\text{p2}}} w(L, ic)\ T_{\text{ex}}(L, ic)\text{dt}\ , \tag{3.12-4}$$

where

$w(L, ic, t = t_{\text{p2}})$ = computed flow rate at t_{p2}

$T_{\text{ex}}(L, ic)$ = coolant temperature at the subassembly inlet or outlet

Additionally, the core coolant routines supply the coefficients $C_0'(L, ic)$, $C_1(L, ic)$, $C_2(L, ic)$, and $C_3'(L, ic)$ used by PRIMAR-4 to estimate the core channel flow. PRIMAR-4 estimates the flow into or out of each subassembly using

$$\begin{aligned}\frac{\Delta w_{\text{e}}(L, ic)}{\Delta \text{t}} = C_0'(L, ic) &+ C_1(L, ic)\left(p(\text{JIN}) + \theta_{2c}\Delta p(\text{JIN})\right) \\ &+ C_2(L, ic)\left(p(\text{JX}) + \theta_{2c}\Delta p(\text{JX})\right) \\ &+ 2\theta_{2c} C_3'(L, ic)\, w_{\text{e}}(L, ic)\, \Delta w_{\text{e}}(L, ic)\end{aligned} \tag{3.12-5}$$

The core channel calculations use w as the flow rate per pin, whereas PRIMAR-4 estimates the total flow represented by a channel, so $N_{\text{ps}}\,(ic)$, the number of pins per subassembly times the number of subassemblies represented by the channel, comes into Eq. (3.12-3), Eq. (3.12-4) and the calculations of the coefficients C_0', C_1, C_2, and C_3'. In the pre-voiding module the coefficients are calculated as

$$\begin{aligned} C_0' = -\frac{N_{ps}}{I_1}\Big(& w_2^2 I_2 + A'_{\text{fL}} w_2 I_{3\text{L}} + A'_{\text{fr}} w_2 \left|w_2\right|^{1+b_{\text{fr}}} I_{3\text{T}} + w_2 \left|w_2\right| I_4 \\ & + g\left[I_5 + \rho_{\text{cin}}\left(z_{\text{c}}\left(1\right) - z_{\text{pll}}\right) + \rho_{\text{cout}}\left(z_{\text{plu}} - z\left(\text{MZC}\right)\right)\right]\Big) \end{aligned} \tag{3.12-6}$$

$$C_1 = \frac{N_{\text{ps}}}{I_1} \tag{3.12-7}$$

$$C_2 = -C_1 \tag{3.12-8}$$

and

$$C_3' = -\frac{\left(I_2 w_2 + 0.5 A'_{\text{fL}} I_{3\text{L}} + \left(1 + 0.5 b_{\text{fr}}\right) A'_{\text{fr}} I_{3\text{T}} \left|w_2\right|^{1+b_{\text{fr}}} + I_4 \left|w_2\right|\right)}{\left|w_2\right| I_1 N_{\text{ps}}} \tag{3.12-9}$$

In this case, the coefficients for $L = 2$ are equal to those for $L = 1$.

3.13 Subroutine Descriptions and Flowcharts

The subroutines used in the core-channel thermal hydraulics calculations are described below, grouped by the phase of the calculation where they are used.

Table 3.13.1: Steady-state Thermal Hydraulics Routines

Routine	Description
SSTHRM	Driver for the steady-state calculations:
SSCOOL	Calculates steady-state coolant, structure, reflector, and gas plenum temperature for each axial node. Also calculates coolant pressure and saturation temperature at each axial node.
SSCLM1	Calculates steady-state coolant pressures and saturation temperatures for the multiple pin option.
SSHTR	Calculates steady-state fuel and cladding temperatures in the core and blankets.
SSNULL	Driver for the steady-state null transient.

Table 3.13.2: Pre-Voiding Transient Temperature Calculations

Routine	Description
TSCL0	Driver of the pre-voiding thermal hydraulics.
TSHTRN	Driver for the pre-voiding transient temperature calculations. Calls TSHTN1, TSHTN2, TSHTN3, TSHTN4, TSHTM2, and TSHTM3 as appropriate.
TSHTN1	Calculates reflector, coolant, and structure temperatures in a reflector zone.
TSHTN2	Calculates cladding, coolant, structure, and plenum gas temperatures in the gas plenum region.
TSHTN3	Calculates fuel, cladding, coolant, and structure temperatures in the core and axial blankets.
TSHTN4	Calculates the axial node size and the coolant density and specific heat for each axial node.
TSHTN5	Called from TSHTN3 to adjust fuel and cladding temperatures to account for the heat of fusion if melting is occurring.
TSHTM2	Multiple pin version of TSHTN2.
TSHTM3	Multiple pin version of TSHTN3.

Table 3.13.3: Fuel and Cladding Temperatures During Boiling

Routine	Description
TSBOIL	Driver for the boiling module.
TSHTRV	Calculates fuel and cladding temperatures in the core and axial blankets.

Table 3.13.4: Pre-voiding Coolant Flow Rates and Pressures

Routine	Description
TSCNV1	Extrapolates coolant temperatures, computes coolant flow rate and pressures. Also tests for start of boiling. Sets variables for coupling with PRIMAR-4. Called by TSCLO every coolant step.
TSCNV2	Called by TSCNV1 to compute flow-rate coefficients x_{I1}(JC), x_{I2}(JC), x_{I3L}(JC), x_{I3T}(JC), x_{I4}(JC) and x_{I5}(JC).
TSCNV3	Called by TSCLO, only at the end of a heat-transfer step, to get heat-transfer coefficients H_{erc}(JC) and H_{sic}(JC). Calls TSCNV8 to get liquid heat-transfer coefficients h_c(JC).
TSCNV7	Computes ρ_c, μ_c for liquid sodium.
TSCNV8	Computes liquid heat-transfer coefficient h_c (JC).
TSCLM1	Multiple pin version of TSCNV1.

Table 3.13.5: Auxiliary Routines

Routine	Description
CFUEL	Calculates the specific heat of the fuel as a function of temperature. For coding efficiency, one call to CFUEL gives the specific heats for all radial nodes at an axial node, rather than using as separate call to CFUEL for each radial node of each axial node.
CCLAD	Calculates the specific heat of the cladding as a function of temperature. One call to CCLAD returns the values for all axial nodes in the core and axial blankets.
KFUEL	Calculates the thermal conductivity of the fuel as a function of temperature. One call to KFUEL provides the values for all radial nodes at an axial node.
KCLAD	Calculates cladding thermal conductivity as a function of temperature. One call to KCLAD returns the values for all axial nodes in the core and axial blankets.
INTRP	General interpolation routine that uses linear interpolation from a table to obtain y as a function of x. A single call to INTRP with an array of x's will return an array of resulting y's.
SHAPE	Multiplies the array of axial power shapes by the current time-dependent power to obtain the current power, in watts, at each axial node in a channel.
HBSMPL	Calculates the bond gap conductance at each axial node if the simple bond gap conductance model is used. Otherwise, DEFORM-4 calculates the bond gap conductance; and HBSMPL is not called.
INVRT3	Solves a tri-diagonal matrix equation of arbitrary size.

Figure 3.1.2 is a flowchart for subroutine TSCLO, Figure 3.13.1 is a flowchart for subroutine TSHTRN, and Figure 3.13.2 is a flowchart for subroutine TSHTN3. The logic in the other pre-voiding single pin transient heat-transfer routines is very simple and straightforward. Subroutine TSHTRV is similar to TSHTN3, except that TSHTRV does not calculate coolant and structure temperatures. Also, in TSHTRV the axial nodes are completely de-coupled, so the order in which they are treated is immaterial. For simplicity, the axial node loop in TSHTRV always starts at the bottom and works up.

3.14 Subchannel Treatment Details

Argonne and the Korea Atomic Energy Research Institute (KAERI) cooperated on an International Nuclear Energy Research Initiative on "Passive Safety Optimization in Liquid Sodium-Cooled Reactors." Part of this work involved the development, implementation, and testing of a detailed whole core coolant subchannel model for inclusion in the Argonne SAS4A/SASSYS-1 LMR safety analysis code and the KAERI SSC-K systems code [3-8]. The purpose of this work was to increase the accuracy and decrease the uncertainties in calculations of reactor safety margins in accident situations. Steady-state and transient hot channel factors are computed mechanistically with a detailed subchannel model for the core, coupled with a thermal hydraulic treatment of the primary and intermediate heat transport loops. It should be noted here that some of the notation used in this section does not match that used in the preceding sections of this chapter. Therefore, all variables are defined in order to avoid confusion.

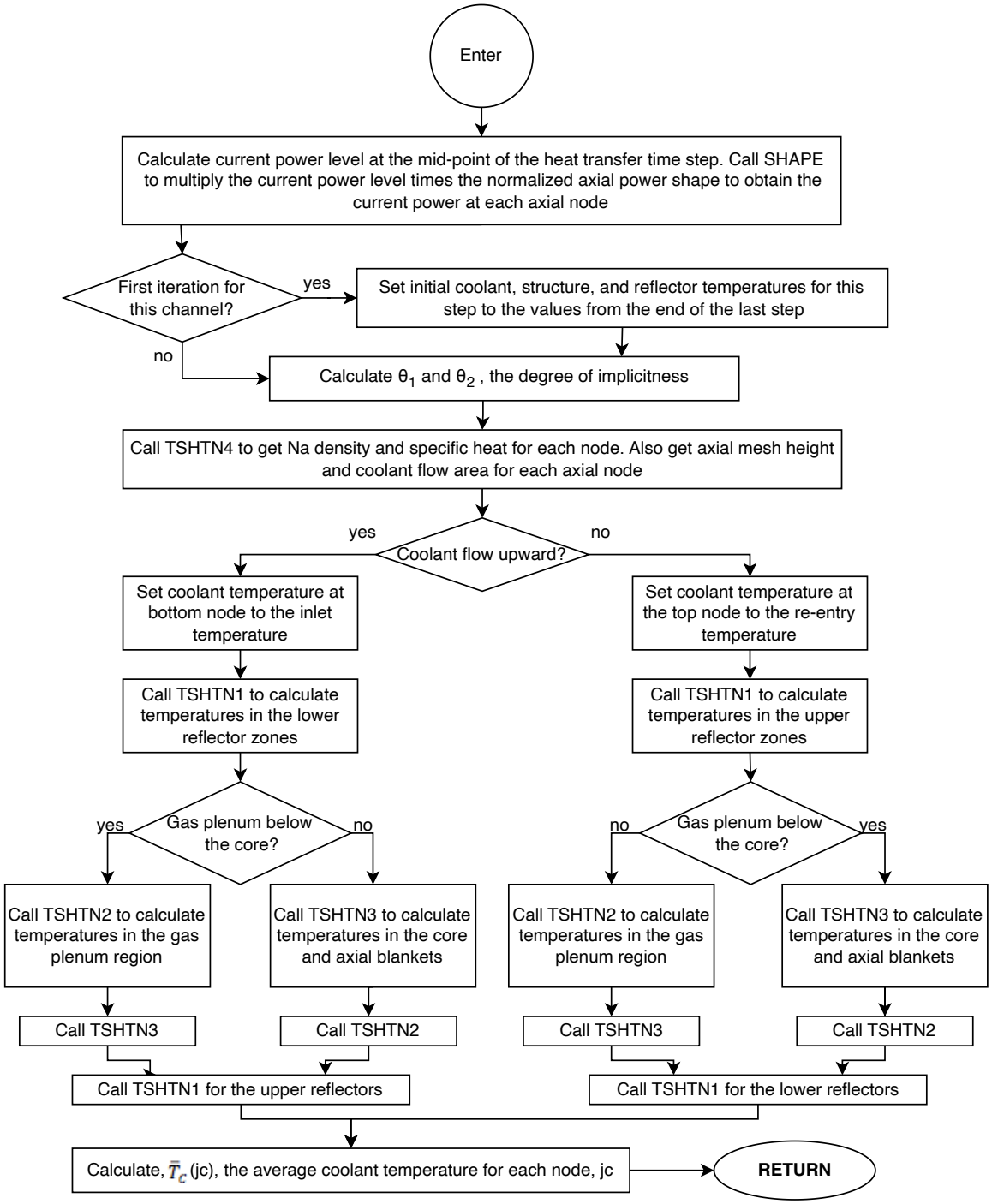


Figure 3.13.1: Flowchart for Subroutine TSHTRN

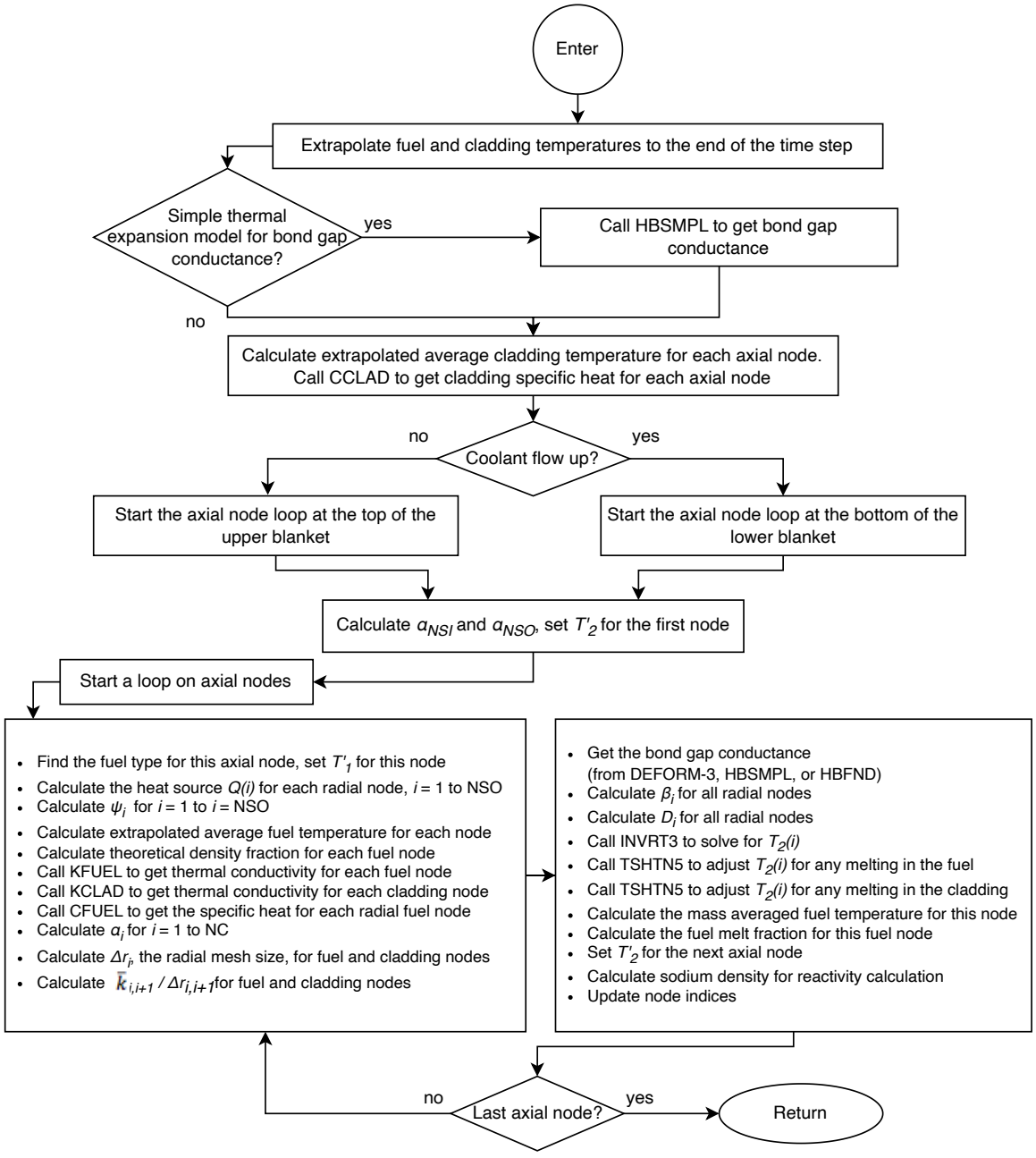


Figure 3.13.2: Flowchart for Subroutine TSHTN3

3.14.1 Model Features

The new model uses a coolant subchannel treatment similar to that used by the COBRA-4 code [3-9] or the SUPERENERGY-2 code [3-10]. The subchannel treatment includes axial coolant flow parallel to the pins and cross flow between coolant subchannels driven by pressure differences and wire wrap sweeping. Heat flow between adjacent coolant subchannels is calculated, including effects of turbulent mixing. The channel treatment includes the whole length of the subassembly, with the detailed subchannel treatment in the pin section and a simpler treatment above and below the pins.

Figure 3.14.1 shows a typical subchannel treatment for fuel pins with a triangular arrangement in a hexagonal duct. In this arrangement, there are interior subchannels, edge subchannels and corner subchannels. If the pins are wrapped with spacer wires, then there would be wire wrap flow sweeping along the duct walls in the edge and corner subchannels. This is the type of geometry the model is intended for, but the geometry is not hard-wired into the code. The model is flexible enough to model other geometries, such as pins in a square arrangement with grid spacers and no wire wraps. The new model couples with existing models for the remainder of the primary loop, intermediate loop and balance of plant.

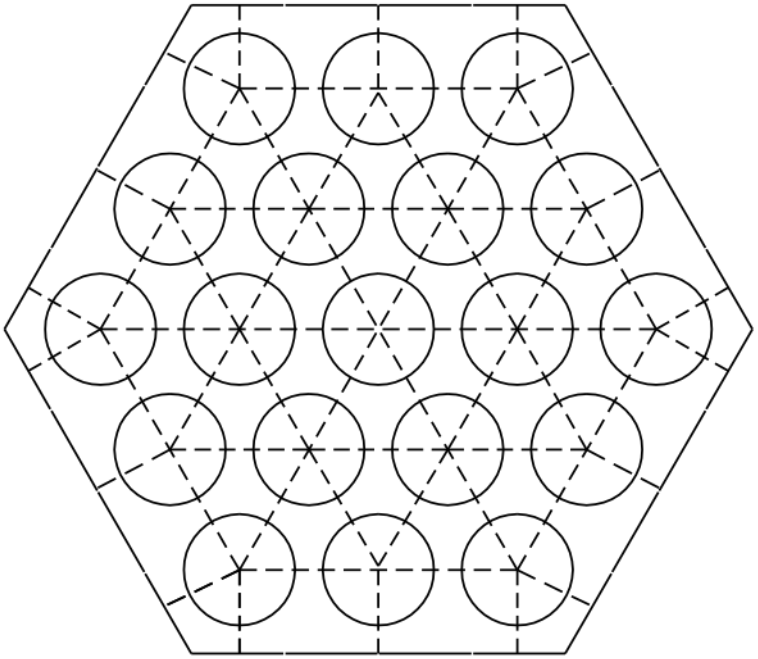


Figure 3.14.1: Coolant Subchannels in a 19 Pin Hex

3.14.2 Basic Equations

Figure 3.14.2 shows the main coolant subchannel variables used in the model for an axial node.

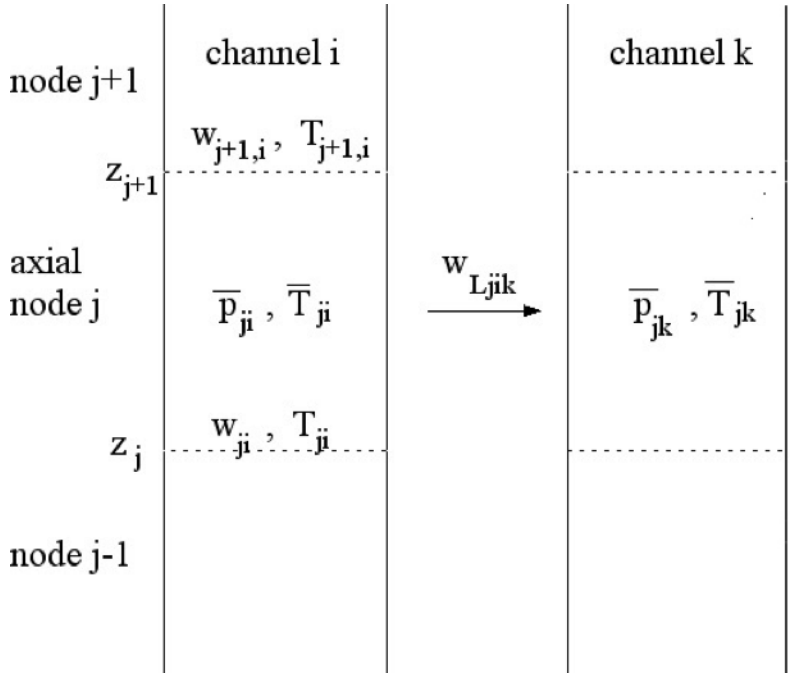


Figure 3.14.2: Coolant Subchannel Variables

The coolant variables in Figure 3.14.2 are defined as:

$\overline{p}_{ji}$ = coolant pressure at the middle of node j in channel i

$\overline{T}_{ji}$ = average coolant temperature for node j in channel i

w_{ji} = coolant flow rate in the axial direction at the bottom of node j in sub- channel i

w_{Ljik} = lateral flow rate from subchannel i to subchannel k at node j

z_j = axial location at the bottom of node j

The basic continuity equation for the coolant in an axial node of a subchannel is

$$\frac{d}{dt}(\overline{\rho}_{ji}A_{ji}\Delta z_j) = w_{ji} - w_{j+1,i} - \sum_k w_{Ljik} \tag{3.14-1}$$

The coolant momentum equation is

$$\begin{aligned}\frac{1}{2}\left(\frac{\Delta z_j}{A_{ji}} + \frac{\Delta z_{j-1}}{A_{j-1,i}}\right)\frac{dw_{ji}}{dt} &= \overline{p}_{j-1,i} - \overline{p}_{ji} - \Delta p_{frji} - \frac{w_{ji}|w_{ji}|K_{orji}}{2\rho_{ji}A_{ji}^2} - \rho_{ji}g(\Delta z_j + \Delta z_{j-1})/2 \\ &- \frac{\overline{w}_{ji}^2}{\overline{\rho}_{ji}A_{ji}^2} + \frac{w_{ji}^2}{\rho_{ji}A_{ji}^2} - \frac{w_{ji}^2}{\rho_{ji}A_{j-1,i}^2} + \frac{\overline{w}_{j-1,i}^2}{\overline{\rho}_{j-1,i}A_{j-1,i}^2} - \frac{1}{2}\sum_k[(1-S_{wjik})w_{Ljik}\left(\frac{w_{jk}}{\overline{\rho}_{jik}A_{ji}A_{jk}}\right. \\ &\left. - \frac{w_{ji}}{\overline{\rho}_{jik}A_{ji}^2}\right) + (1-S_{wj-1,ik})w_{Lj-1,ik}\left(\frac{w_{j-1,k}}{\overline{\rho}_{j-1,ik}A_{j-1,i}A_{j-1,k}} - \frac{w_{j-1,k}}{\overline{\rho}_{j-1,ik}A_{j-1,i}^2}\right)]\end{aligned} \tag{3.14-2}$$

The energy equation for the coolant is

$$V_{\text{cji}} C_{\text{prefji}} \frac{\text{d}}{\text{dt}} [\overline{\rho}_{\text{ji}} (\overline{T}_{\text{ji}} - T_{\text{r}})] = \sum_{\text{in}} w_{\text{inji}} (T_{\text{inji}} - T_{\text{r}}) C_{\text{prefji}} - \sum_{\text{out}} w_{\text{outji}} (T_{\text{out}} - T_{\text{r}}) C_{\text{prefji}} + \phi_{\text{ji}} \tag{3.14-3}$$

The energy equation in the fuel is

$$\rho_{\text{f}} C_{\text{f}} \frac{\text{dT}_{\text{f}}}{\text{dt}} = \frac{1}{r} \frac{\text{d}}{\text{dr}} (k_{\text{f}} r \frac{\text{dT}_{\text{f}}}{\text{dr}}) + Q \tag{3.14-4}$$

A similar equation is used for the cladding, and a bond gap conductance is used between the fuel outer surface and the cladding inner surface. In the above equations:

A = coolant flow area in the subchannel

C_{f} = heat capacity of the fuel

C_{pref} = coolant heat capacity, evaluated at T_{r}

g = acceleration of gravity

i = subchannel number

j = axial node number

k = subchannel number of a connecting subchannel

k_{f} = fuel thermal conductivity

K_{or} = orifice coefficient

Δp_{frji} = friction pressure drop in the bottom half of node j and the top half of node j-1

Q = heat source per unit volume in the fuel

S_{wjik} = 1 if $w_{\text{ljik}} \geq 0$,

0 otherwise

t = time

T_{f} = fuel temperature

T_{in} = temperature of coolant entering the node, either from another axial node in the same subchannel or from a connecting subchannel

T_{out} = temperature of coolant leaving the node

T_{r} = reference temperature, $\overline{T}$, at the beginning of the time step is used for T_{r}.

V_{c} = coolant volume in the axial node

z = axial distance

ρ_{j} = coolant density at the bottom of node j

$\overline{\rho}_{\text{j}}$ = coolant density at the middle of node j

ρ_{f} = fuel density

ϕ_{ji} = heat source to the coolant node

$$\phi_{\text{ji}} = \phi_{\text{cji}} + \phi_{\gamma\text{ji}} + \phi_{\text{scji}} + \phi_{\text{chchji}} \tag{3.14-5}$$

ϕ_{cji} = heat source from the cladding and structure surfaces to the coolant

$\phi_{\gamma ji}$ = direct heat source to the coolant from neutrons and gamma rays

ϕ_{scji} = heat source from turbulent mixing and conduction from adjacent sub- channels

ϕ_{chchj} = heat source to the coolant from channel-to-channel heat transfer

Also,

$$\phi_{scji} = \sum_k [u_{1ik} k_j + u_{2ik} C_j (\overline{w}_{ji} + \overline{w}_{jk})](\overline{T}_{jk} - \overline{T}_{ji}) \Delta z_j \tag{3.14-6}$$

where

u_{1ik} = geometry factor for conduction from subchannel i to subchannel k

u_{2ik} = turbulent mixing factor

Δz_j = axial node length

The parameters u_{1ik} and u_{2ik} are geometry-dependent factors supplied by the user for the particular geometry being used.

Based on a numerical simulation [3-11] with the CFX code for a triangular array of rods, the conduction geometry factor can be obtained from

$$u_{1ik} = 0.7774 \left(\frac{s}{L_c}\right) \left(\frac{P}{D}\right) \left(\frac{s}{D}\right)^{-0.2627} \Delta z_j \tag{3.14-7}$$

where

s = gap size between subchannels

L_c = centroid distance between subchannels

This correlation only accounts for thermal conduction in the coolant. It may be desirable to increase u_{1ik} to account for circumferential thermal conduction in the clad and possibly conduction in the fuel.

The value for u_{2ik} can be obtained from the parameter ϵ^* used by Cheng and Todreas [3-12].

$$u_{2ik} = \frac{\epsilon^*_{i\eta} c}{(A_{ci} + A_{ck})} \tag{3.14-8}$$

$c = s$ = gap width

A_{ci} = coolant flow area in subchannel i

The coolant lateral flow rate is given by

$$w_{Ljik} = K_{sik} \Delta z_j \frac{(w_{ji} + w_{j+1,i})}{2} + w_{Lpjik} \tag{3.14-9}$$

K_{sik} = wire wrap sweeping factor

In this equation the first term is the net flow due to wire wrap sweeping, and the second term is the pressure driven subchannel-to-subchannel flow. Note that K_{sik} = 0 unless there is net sweeping from subchannel i to subchannel k. For a gap between an inner subchannel and another inner subchannel or between an inner subchannel and an edge subchannel there are two wrapper wires involved; one on each pin. The two wires sweep in opposite directions, so the net sweep flow is approximately zero. Normally the wire wrap sweeping factor is zero unless i and are both edge or corner subchannels.

The second term in Eq. (3.14-9) is determined by

$$\left(\frac{L}{A}\right)_{Lat} \frac{dw_{Lpjik}}{dt} = \overline{p}_{ji} - \overline{p}_{jk} - \frac{w_{Lpjik}|w_{Lpjik}|}{2\overline{\rho}_{jik} A_L^2} K_{Lat} \text{ if } |\overline{p}_{ji} - \overline{p}_{jk}| > \Delta p_{lam} \tag{3.14-10}$$

or

$$\left(\frac{L}{A}\right)_{\text{Lat}} \frac{\text{dw}_{\text{Lpjik}}}{\text{dt}} = \overline{p}_{\text{ji}} - \overline{p}_{\text{jk}} - K_{\text{Lat}} \frac{w_{\text{lam}} w_{\text{Lpjik}}}{2\overline{\rho}_{\text{jik}} A_{\text{L}}^2} \text{ otherwise} \tag{3.14-11}$$

where

K_{Lat} = effective lateral flow orifice coefficient

$\left(\frac{L}{A}\right)_{\text{Lat}}$ = lateral inertia term

Δp_{lam} = user specified transition pressure difference

and

$$w_{\text{lam}} = \frac{2\overline{\rho}_{\text{jik}} A_{\text{L}}^2}{K_{\text{Lat}}} \Delta p_{\text{lam}} \tag{3.14-12}$$

K_{sik} is supplied by the user. It can be obtained from the parameter $C_{1\text{L}}$ used by Cheng and Todreas:

$K_{\text{sik}} = cC_{1\text{L}}/A_{\text{c}}$

Cylindrical geometry with radial heat conduction is used for calculating the fuel pin temperatures. Figure 3.14.3 shows the radial node structure used for a fuel pin.

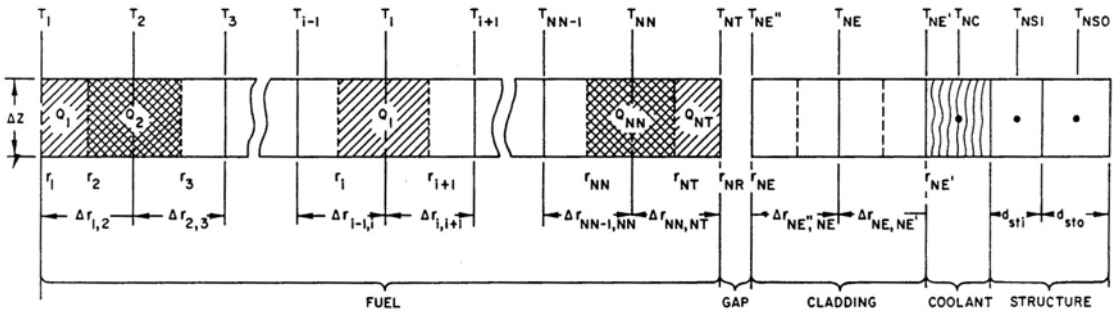


Figure 3.14.3: Radial Node Structure Used for a Fuel Pin, with Coolant and Structure

For an interior fuel node, $1 < i <$ NT, the conduction equation (equation Eq. (3.14-4)) becomes:

$$m_{\text{fi}} C_{\text{fi}} \frac{\text{dT}_{\text{fi}}}{\text{dt}} = 2\pi r_{\text{i+1}} \overline{k}_{\text{i,i+1}} (T_{\text{i+1}} - T_{\text{i}}) + 2\pi r_{\text{i}} \overline{k}_{\text{i-1,i}} (T_{\text{i-1}} - T_{\text{i}}) + Q_{\text{i}} \tag{3.14-13}$$

where

m_{fi} = fuel mass per unit height in node i

Q_{i} = heat source per unit height in node i

$\overline{k}_{\text{i,i+1}}$ = effective average thermal conductivity for heat flow from node i to i+1

Similar equations are used for node 1, node NT and the cladding.

3.14.3 Numerical Procedures

The solution methods used for the subchannel treatment are dictated by stability requirements and by the features that the model includes. In the coolant, a compressible treatment is desired to provide the pressures to drive subchannel-to-subchannel flow rates. Eq. (3.14-1) above implies a compressible treatment. With a compressible treatment and using explicit time differencing, the Courant limit would apply. The sonic speed in sodium is about 2400 m/s. A typical axial node size is 0.03 m. Thus the Courant limit would restrict the time step size to about 10^{-5} seconds or less. To avoid such a small limit, an implicit solution is used. For one time step, the pressures and flows in all channels representing a subassembly are solved for simultaneously.

For the coolant temperatures a similar limit applies. The stability limit for an explicit solution is given by the axial node size divided by the coolant velocity. A node size of 0.03 m and a coolant velocity of 7 m/s corresponds to a limit of about 0.004 seconds. Thus, implicit time differencing is also used in this case and the coolant temperatures in a subassembly are solved for simultaneously. Also, it is probably necessary to solve for coolant temperatures simultaneously with the pressures and flows in order to obtain a stable solution. Therefore, all coolant temperatures, pressures, and flows in a subassembly are solved for simultaneously. Fortunately the situation with subassembly-to-subassembly heat transfer is different. In the PRISM design the hex can wall thickness is 0.0039 m, and the duct gap is 0.0043 m, giving a stability limit of about 2.6 seconds for explicit can wall-to-can wall heat transfer. Therefore, the subassembly-to-subassembly heat transfer can be treated explicitly, using conditions at the beginning of the time step. This means that for a given time step each subassembly can be treated separately using time step sizes of 1.0 second or more.

The steady-state calculation begins with an initial approximation of the steady-state coolant temperatures, pressures, and flow rates for each subassembly where the heat flux from each fuel pins to the coolant is determined by the steady-state pin power. Then a null transient is run for each subassembly separately, neglecting subassembly-to-subassembly heat transfer. Finally, a null transient with subassembly-to-subassembly heat transfer is run for all subassemblies. During both null transients the powers and subassembly coolant inlet flows are held constant. After the coolant conditions are calculated, the fuel pin and structure temperatures are calculated.

A time step approach is used both for the steady-state null transient and for the regular transient. Conditions are known at the beginning of the time step, and the main computational task is to determine the conditions at the end of the step. The equations are linearized about values at the beginning of the time step, and fully implicit finite differencing in time is used for the basic conservation equations. This leads to N linear equations in N unknowns. The unknowns are solved for by iteration. Explicit time differencing is used for the subassembly-to-subassembly heat transfer, so the calculations for one time step for each subassembly can be done independently.

For a transient time step the heat flux from the fuel pins and structure to the coolant is approximated as

$$\phi_{\mathrm{cji}} = \phi_{\mathrm{c1ji}} + \phi_{\mathrm{c2ji}} \Delta \overline{T}_{\mathrm{ji}} \tag{3.14-14}$$

for the coolant calculations. The coefficients ϕ_{c1} and ϕ_{c2} are calculated in the fuel pin heat transfer routines. In this equation ΔT is the change in coolant temperature during the time step. After the coolant temperatures are calculated, the fuel pin and structure temperatures are calculated for the time step.

3.14.4 Interfacing with Other Models in the Code

This section summarizes the interfacing between the existing SAS4A/SASSYS-1 code and the new three-dimensional thermal hydraulics core model. The main interactions between the new model and the rest of the code are with the input module, output module, neutronics module, DEFORM-5 fuel pin mechanics module for metal fuel, simple PRIMAR-1 primary loop module, and detailed PRIMAR-4 module for the primary and intermediate heat transfer loops. These interactions and the interface data requirements are discussed in the sections below.

Note that the new subchannel model is only applicable to single-phase coolant with intact fuel pins. The SAS4A/SASSYS-1 sodium boiling model and PLUTO and LEVITATE disrupted fuel pin modules cannot be used in the same subassembly as the coolant subchannel module.

3.14.4.1 Input Module

The SAS4A/SASSYS-1 code reads input into numbered locations in input blocks. Extra space has been provided in these input blocks for future use by new modules. Some of this extra space is used by the new input variables for the new three dimensional thermal hydraulics core module. The new module also uses many existing input variables. There is enough extra space in each input block to accommodate the new variables. No modifications are required to the input module for use with the new module. The new input variables are listed in the user guide section below.

3.14.4.2 Output Module

The output files for a whole-core case using the new model can get very large because of the large number of channels involved. Therefore, some modification of the output module has been necessary. For the printed output one change that was necessary was to print reactivity feedback by subassembly rather than by channel. Output on CHANNEL.dat is binary data intended for input to plotting programs. Previously one record per channel per time step was put on CHANNEL.dat. This has been modified to output every N time steps for specified channels or subassemblies.

3.14.4.3 Neutronics Module

The neutronics module uses fuel, cladding, and coolant temperatures and coolant densities from the core thermal hydraulics module to obtain reactivity feedback components at the end of each transient main time step. For each transient step, the neutronics module supplies the core thermal hydraulics module with the power level. The coupling with the neutronics module is the same for the new three dimensional thermal hydraulics core module as for the older core thermal hydraulics module.

3.14.4.4 DEFORM-5 Fuel Pin Mechanics for Metal Fuel

The DEFORM-5 module uses fuel and cladding temperatures from the core thermal hydraulics module for its prediction of pin failure. The coupling with the DEFORM-5 module is the same for the new three-dimensional thermal hydraulics core module as for the older core thermal hydraulics module.

3.14.4.5 PRIMAR-1

SAS4A/SASSYS-1 uses either the simple PRIMAR-1 primary loop module or the detailed PRIMAR-4 module for the primary and intermediate heat transport systems. When PRIMAR-1 is used, the initial steady-state subassembly inlet pressure and average core outlet temperature calculated by the core thermal hydraulics module are used by PRIMAR-1 to obtain the initial steady-state primary loop gravity head and pump head. During the transient calculation, PRIMAR-1 provides the subassembly inlet pressure and inlet temperature to the core thermal hydraulics module. No transient information from the core thermal hydraulics module is used by PRIMAR-1.

3.14.4.6 PRIMAR-4

The core thermal hydraulics module and PRIMAR-4 are tightly coupled, and the transient coupling with PRIMAR-4 required more effort in the development of the new core thermal hydraulics module than the coupling with any other module. This coupling also contributes significantly to the running time of the new module.

The steady-state coupling between the core thermal hydraulics and PRIMAR-4 is similar to the coupling with PRIMAR-1. The user specifies the subassembly outlet pressure and inlet temperature in the input. The initial coolant flow rate in each channel is also specified by the user. Then the core thermal hydraulics module calculates the outlet temperature and inlet pressure for each subassembly. Inlet orifice coefficients are adjusted so that the inlet pressures for all subassemblies match the maximum inlet pressure. The subassembly inlet and outlet pressures, inlet temperatures and average outlet temperature are used in the steady-state PRIMAR-1 and PRIMAR-4 calculations. This steady-state coupling is the same with the new core thermal hydraulics model.

During the transient calculation, PRIMAR-4 provides the subassembly inlet pressure and inlet temperature to the core thermal hydraulics module in the same way that PRIMAR-1 does. The big difference between the coupling with PRIMAR-4 and PRIMAR-1 is that during a transient time step, PRIMAR-4 estimates the core flows and outlet temperatures before they are calculated by the core thermal hydraulics module. With the previous thermal hydraulics module at the end of each PRIMAR time step, the core module calculates the coefficients $C_0, C_1, C_2,$ and C_3 for each channel for use by PRIMAR-4. At the beginning of the next PRIMAR time step, PRIMAR-4 gets the channel flow rates from the core thermal hydraulics module and estimates the channel flow at the end of the step using

$$\frac{\mathrm{dw_{Li}}}{\mathrm{dt}} = C_{0\mathrm{Li}} + C_{1\mathrm{Li}} p_{\mathrm{in}} + C_{2\mathrm{Li}} p_{\mathrm{x}} + C_{3\mathrm{Li}} w_{\mathrm{Li}} |w_{\mathrm{Li}}| \tag{3.14-15}$$

where

w = coolant flow rate

t = time

L = 1 for the subassembly inlet, 2 for the subassembly exit

i = channel number

With the new core thermal hydraulic treatment, at the beginning of each transient PRIMAR-4 time step, after the time step size has been determined, the core thermal hydraulics module is called to provide new coefficients $a_0,\ a_1$ and a_2 for each subassembly. PRIMAR-4 then estimates the coolant flow at the end of the PRIMAR step using

$$w_{\mathrm{Ln}}\left(t_{\mathrm{p}} + \Delta t\right) = w_{\mathrm{Ln}}\left(t_{\mathrm{p}}\right) + \left(a_{0\mathrm{Ln}} + a_{1\mathrm{Ln}} \Delta p_{\mathrm{in}} + a_{2\mathrm{Ln}} \Delta p_{\mathrm{x}}\right) \Delta t \tag{3.14-16}$$

where

n = subassembly number

t_{p} = time at beginning of the PRIMAR step

Δt = PRIMAR time step size

Δp_{in} = change in inlet pressure during the PRIMAR step

Δp_{x} = change in outlet plenum pressure during the PRIMAR step

Note that with the proper choice of values for the coefficients, Eq. (3.14-15) and Eq. (3.14-16) can be made approximately equivalent, but there are two significant differences between them. First when using the previous core thermal hydraulics module, the C's are calculated for each channel, based on the assumption that one channel represents one or more subassembly. With the new core thermal hydraulics module, a number of channels can be used to represent one subassembly, and the a's of Eq. (3.14-16) are calculated for each subassembly rather

than for each channel. Second, the coefficients in Eq. (3.14-15) are calculated before the new PRIMAR time step size is known; and the coefficients are independent of time step size. On the other hand, the coefficients in Eq. (3.14-16) are calculated after the new PRIMAR time step size is known, and they may depend on the size of the time step. The previous core thermal hydraulics module uses incompressible coolant flow, whereas the new module uses a compressible treatment for the coolant. With an incompressible treatment, it is possible to obtain C's for Eq. (3.14-15) that are approximately correct for any reasonable time step size. On the other hand, with a compressible treatment the values of the coefficients in Eq. (3.14-16) will depend strongly on whether or not the time step is large enough for a sound wave to travel from one end of the subassembly to the other and back in the time step.

3.14.5 Data Management

A very flexible data management scheme is required for the detailed subchannel model because of the range of size of the cases that will be run and because a lot of the data is needed simultaneously in the solution. The range of sizes of the cases expected to be run with the model is very large: from a single subassembly containing a few pins and a few coolant subchannels to a large whole core case containing hundreds of subassemblies with hundreds of channels per subassembly. Also, in the calculations for a time step the equations for all channels in a subassembly are solved simultaneously, requiring simultaneous access to the data for all of the channels in the subassembly.

In order to provide the flexibility needed by the model, most of the data is stored in dynamically allocated storage containers whose sizes are determined at run time based on the size of the case being run. A structured pointer system is used to access the data. The permanent channel dependent variables, which are saved from one time step to the next, are stored in a large container and accessed with channel-dependent pointers. The temporary variables used in the simultaneous solution for all of the channels in a subassembly are stored in a number of containers also accessed with pointers. Thus small cases can be run with a small amount of memory, and the available memory can be apportioned efficiently to run large cases without re-compiling the code.

3.14.6 Coding Overview and Subroutines

Figure 3.14.4 shows an overview of the steady-state calculations.

Figure 3.14.5 shows an overview of the transient calculations for the model. It should be noted that SSHTR is a previously existing routine in the SAS4A/SASSYS-1 code. Also, TSHTSC is based mainly on the previously existing routine TSHTRV, with some modifications. Listings of the subroutines used in the new model are given in Table 3.14.1, Table 3.14.2, and Table 3.14.3.

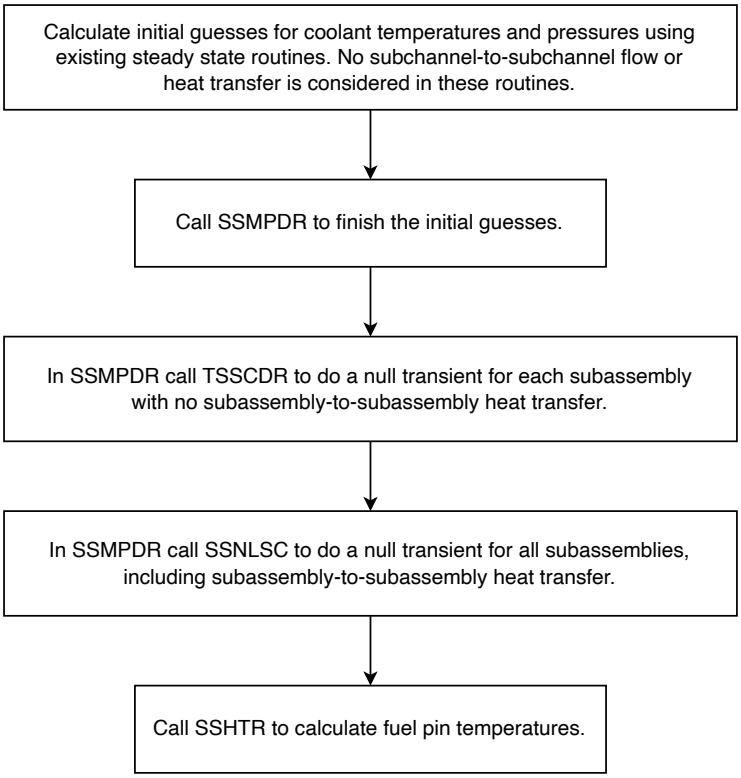


Figure 3.14.4: Coding Overview of the Steady-State Calculation for the Detailed Subchannel Model

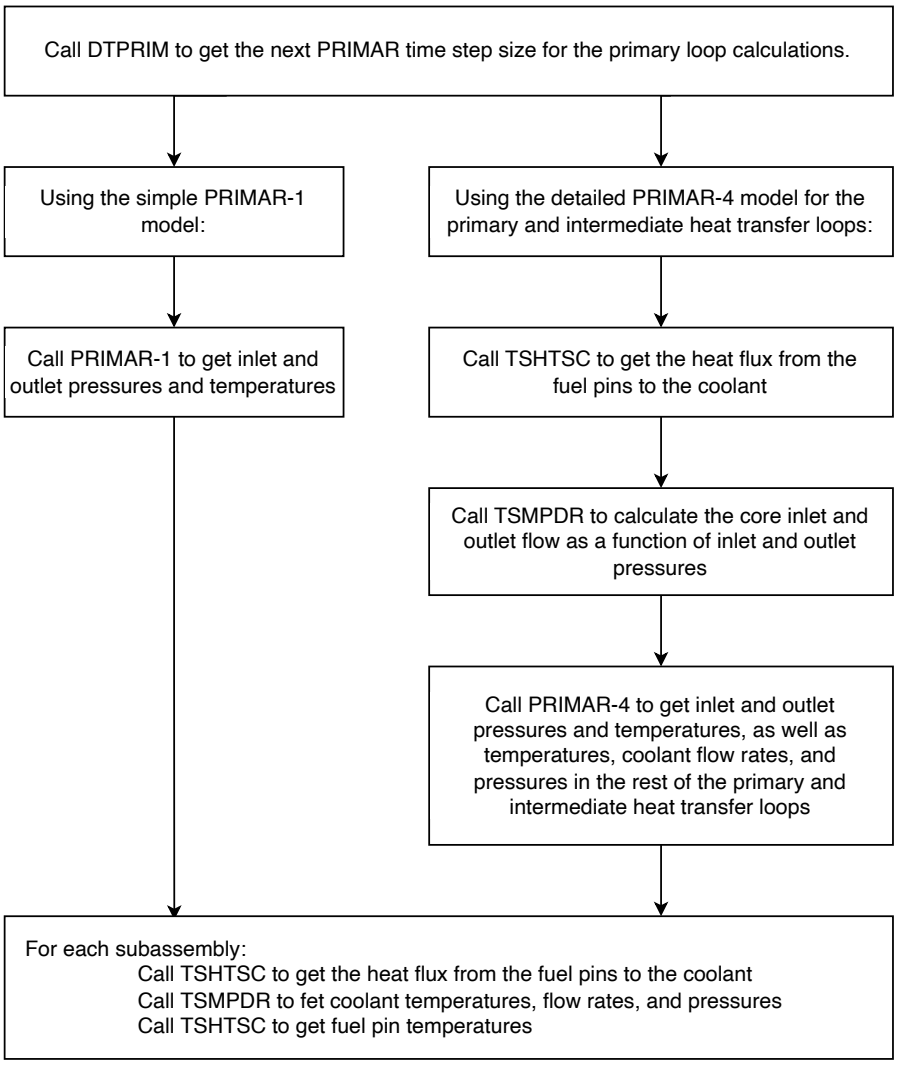


Figure 3.14.5: Coding Overview for One Time Step of the Transient Calculation for the Detailed Subchannel Model

Table 3.14.1: New Subroutines - Detailed Coolant Subchannel Model

Subroutine	Description
CHCHFL	Calculates subassembly-to-subassembly heat flow from the outer structure node of channel i to the outer structure node of channel j. Also contains an option to transfer heat from the outer structure node of channel i to the coolant of channel j. In the future this subroutine will be modified to include an option to transfer heat from the outer structure node of channel i to a constant temperature heat sink. This is a modified version of a SAS4A/SASSYS-1 subroutine.
COOLFN	Finishes the coolant calculations for one time step for one group of channels after SOLVIT has calculated the changes in the main variables.
COOLPT	Prints the coolant results for one channel at the end of a time step.
DYNAL2	Dynamic storage allocation for containers for temporary storage for the coolant model.
INVTRI	Solves a general tri-diagonal matrix equation.
LATPRT	Prints coolant lateral flow rates.
SBFT24	Writes selected binary output on unit 24. Section 3.15.3.1.4 has more detail on the unit 24 output.
SCCOEF	Calculates the coefficients of the linearized equations for the coolant model for one group of channels for one time step.
SOLVIT	Solves the equations set up by SCCOEF to obtain the changes in the coolant independent variables for a time step.
SSMPDR	Steady-state driver for the detailed coolant subchannel model. Calculates initial approximate values for coolant variables. Then runs a null transient for each subassembly without subassembly-to-subassembly heat transfer. Finally runs a null transient for the whole core with subassembly-to-subassembly heat transfer.
SSNLSC	Driver for the steady-state null transient with subassembly-to-subassembly heat transfer.
SSSTR	Calculates steady-state temperatures in the structure and in the reflectors.
TSHTSC	Calculates transient temperatures in the fuel pins. Also calculates structure and reflector temperatures. A modified version of TSHTRV from SAS4A/SASSYS-1.
TSMPDR	Transient driver for the detailed subchannel model. Calculates one coolant time step for one group of channels.

Table 3.14.2: SAS4A/SASSYS-1 Subroutines Used with Detailed Subchannel Model

Subroutine	Description
CCLAD	Cladding heat capacity.
CFUEL	Fuel heat capacity.
DATMOV	Copies data packs from the dynamically allocated memory container to the regular common blocks and moves them back.
HBSMPL	Simple model for the bond gap conductance between the fuel and the cladding.
INVRT3	Solves a tri-diagonal matrix equation for temperatures. Note, INVRT3 and INVTRI have similar uses, but INVTRI can handle a more general tri-diagonal matrix.
KFUEL	Calculates fuel thermal conductivity.
LINES	Counts the number of lines that have been printed and puts page headings in the output.
POINST	Calculates pointers used by DATMOV and in the memory management.
READIN	Reads the input data
RESTAR	Reads or writes a restart file.
SHAPE	Sets the axial power shape in the fuel pin section of a channel.
SSCOOL	Calculates steady-state coolant temperatures and pressures ignoring subchannel-to-subchannel cross flow and heat transfer. Used for initial conditions for the steady-state null transient.
SSHRT	Steady-state fuel pin temperatures.
TSHTN5	Calculates melting of the cladding.
TSHTN6	Calculates melting of the fuel.
TSPRNT	Prints thermal hydraulic output for selected channels.

Table 3.14.3: SAS4A/SASSYS-1 Drivers that Call Detailed Subchannel Model

Subroutine	Description
INPDRV	Input driver, calls READIN
SSTHRM	Steady-state thermal hydraulics driver for the core channel calculations.
TSTHRM	Transient thermal hydraulics driver for the core channel calculations.

3.14.7 Required Computer Capabilities

One question that has been addressed is whether it is practical to run a detailed whole core case on currently available computers. There are two main computer-related considerations. One consideration is whether currently available computers have enough memory for a whole core case. The other consideration is whether computers are fast enough to run a large case in a reasonable amount of time.

3.14.7.1 Memory Requirements

For the detailed subchannel model the SAS4A/SASSYS-1 code requires 0.13 or 0.17 MB of main memory per channel, depending on whether or not the fuel pin mechanics model in the code is used in addition to the subchannel thermal hydraulics model. An additional 8 MB are used for coding, and a few MB are used for non-channel dependent memory.

To determine how many channels are required for a detailed treatment, consider the ALMR mod B reactor design, which is a moderately sized reactor, with a power rating of 840 MWt. This reactor contains 108 driver subassemblies with 271 pins and 546 coolant subchannels per subassembly, 84 blanket subassemblies with 127 pins and 258 coolant subchannels per subassembly, 66 shield subassemblies with 7 pins and 18 coolant subchannels per subassembly, and a few other subassemblies. With a one computational channel per subchannel treatment, a whole core case could therefore use about 81,000 channels. This would require 10.5 - 13.8 GB of memory, although if the core loading is symmetrical the number of computational channels and the memory requirements could be reduced by making use of the symmetry. Liquid metal reactor designs about four times as large as the ALMR reactor have been considered. A detailed analysis of such a design might require 40 - 50 GB of memory. Some current computers are limited to a maximum of 2 GB of memory, although many computers have more memory. Accessing more than 2 GB of memory would probably require the use of 64 bit integers.

3.14.7.2 Speed Considerations

Timing data for the new detailed subchannel model indicates that the time required on an old Sun Blade computer with a processor speed of 500 MHz is about 8 ms per time step per channel. Thus, even if this computer had enough memory to run the 81,000 channel ALMR case discussed above, the time required for a solution for 1000 time steps could be 650,000 seconds, or 7.5 days. Computers with processor speeds significantly greater than 500 MHz are currently available, but it appears that running a large case will require significant amounts of computing power and running time.

3.14.7.3 Use of a Multi-Processor Cluster

Because of the memory and speed considerations listed above, it might appear that the most practical way to run large cases with the new model is with the use of a cluster of processors, with one or more subassemblies run on each processor. The model was developed with that in mind. However, the older modules in SAS4A/SASSYS-1 were written before clusters of processors were being considered, and converting the whole code to run on multiple processors would require a great effort. Currently the approach is to model part of the core with the detailed subchannel model and to model the rest of the core with the older, less detailed models which require considerably less memory and computing time. This approach works well with the current code and current computers. High-end currently available computers have enough memory and speed to run large detailed subchannel cases on a single computer.

3.15 Input and Output

3.15.1 Input Description

The input variables used in the pre-voiding subassembly thermal hydraulics calculations are listed in Table 3.14.1. Some additional comments on some of these input variables are listed below.

3.15.1.1 Per Pin Basis

All of the core channel thermal hydraulic input to SAS4A/SASSYS-1 is on a per pin basis rather than a per subassembly basis. Thus, the initial flow rate, **W0**, is kg/s per pin; and the perimeters **SRFSTZ** and **SER** are perimeters per pin. Also, **ACCZ** is a coolant flow area per pin, and the variables **DZIAB** and **DZIAT** are ratios of inertial lengths to flow areas per pin.

3.15.1.2 Structure/Duct Wall and Wrapper Wires

Typically a SAS4A/SASSYS-1 channel represents an average pin in a subassembly. In this case, the structure normally represents one pin's share of the duct wall, and it may also include the wrapper wire. If there are N pins in the subassembly, then the thickness used for the structure is the actual duct wall thickness, and the perimeter used for the structure is the duct wall perimeter divided by N. The wrapper wire can be either lumped with the cladding or included in the structure. Since wrapper wires are in much better contact with the cladding than with the duct wall, the most accurate treatment of the wrapper wire is probably to lump it with the cladding by increasing the specific heat of the cladding by enough to account for the total heat capacity of the cladding plus the wrapper wire. The cladding dimensions would not be changed.

The problem with lumping the wrapper wires with the duct wall to produce a single "structure" is that typically the duct wall has a much larger ratio of volume to wetted surface area than a wrapper wire, so the wrapper wire temperature will respond much more rapidly than the duct wall temperature to a change in coolant temperature. The heat capacity of the duct wall is considerably greater than the heat capacity of all of the wrapper wires in a subassembly, but the total perimeter of the wrapper wires is greater than the perimeter of the duct wall. If the duct wall and wrapper wires are lumped together, then the thickness of the structure should be determined by the thickness of the duct wall, since the duct wall contains most of the heat capacity. Then the perimeter of the structure should be chosen to conserve the total volume or total heat capacity of the duct wall plus wrapper wires.

A SAS4A/SASSYS-1 channel can be used to represent a central pin rather than an average pin in a subassembly. In this case, the duct wall would probably be ignored, and the "structure: would represent only the wrapper wire. The perimeter of the structure would equal the perimeter of a wrapper wire, and the thickness of the structure would equal one half of the wrapper wire radius in order to conserve volume.

Table 3.15.1: Subassembly Thermal Hydraulics Input Variables

Variable	Reference	Input Variable	Input Block	Location Number	Suggested Value	External Reference
		NCHAN	1	1	1-34	
		IFUEL1	1	3	1-8	
		ICLAD1	1	4	1-3	
		ITKEL	1	7	0 or 1	
		IPOWOP	1	9	1	
		MAXSTP	1	11	–	

continues on next page

Table 3.15.1 – continued from previous page

Variable	Reference	Input Variable	Input Block	Location Number	Suggested Value	External Reference
		IPO	1	12	20-50	
		IPOBOI	1	13	20-50	
		IBLPRT	1	14	0	
		INAS3D	1	29	0	
		ISSNUL	1	87	–	
		IPRSNL	1	88		
		EPSTEM	11	1	.1 or less	
		DTMXB	11	6	.01	
		DTFUEL	11	10	50	
		DTCLAD	11	11	30	
		POW	12	1		
ρ	Section 3.3.1	**COEFDS**(1)	13	1	11080	
		COEFDS(2)	13	2	2.04×10^{-5}	
		COEFDS(3)	13	3	8.70×10^{-9}	
k	Section 3.3.1	**COEFK**	13	4		
k	Section 3.3.1	**EXKTB**	13	11		
		EXKTM	13	71		
ρ	Section 3.3.1	**RHOTAB**	13	91		
		RHOTEM	13	251		
k	Section 3.3.1	**XKTAB**	13	420		
		XKTEM	13	580		
$\bar{c}_f$	Eq. (3.3-17)	**CPFTAB**	13	606		
		CPFTEM	13	766		
T_{sol}	Eq. (3.3-143)	**TFSOL**	13	786		
T_{liq}	Eq. (3.3-144)	**TFLIQ**	13	794		
U_{melt}	Eq. (3.3-3)	**UFMELT**	13	802		
U_{melt}	Eq. (3.3-3)	**UEMELT**	13	816		
c_e	Eq. (3.3-33)	**CPCTAB**	13	819		
$\rho_e c_e$		**CPCTEM**	13	879		
		CROETB	13	990		
		CROETM	13	1050		
p_x	Eq. (3.12-1)	**PX**	14	1		
T_{in}	Eq. (3.3-152)	**T0TAB**	14	45		
		T0TME	14	65		
z_{pll}	Eq. (3.9-43)	**ZPLENL**	14	87		
z_{plu}	Eq. (3.9-39)	**ZPLENU**	14	88		
M_{mix}	Eq. (3.3-150)	**XMXMSI**	14	93		
M_{mix}	Eq. (3.3-150)	**XMXMSO**	14	94		

continues on next page

Table 3.15.1 – continued from previous page

Variable	Reference	Input Variable	Input Block	Location Number	Suggested Value	External Reference
τ_{mix}	Eq. (3.3-152)	TIMMIX	14	95		
		IRHOK	51	3		
		NPLN	51	4	2-6	
		NREFB	51	5	1-5	
		NREFT	51	6	1-5	
		NZNODE	51	7		
NT		NT	51	14	4-11	
		IFUELV	51	15	1 - IFUEL1	
		IFUELB	51	16	1 - IFUEL1	
		ICLADV	51	17	1 - ICLAD1	
		NGRDSP	51	18	0-10	
		IHGAP	51	24	0 or 1	
		NPIN	51	25	1 or more	
		NSUBAS	51	26	1 or more	
		MZUB	51	27	0-24	
		MZLB	51	28	0-24	
		ILAG	51	34	0	
		IEQMAS	51	118	0	
		IFRFAC	51	186		
		NCHCH	51	205		
		ICHCH	51	206		
		JJMLTP	51	214		
		IAXCON	51	284		
A_c	Eq. (3.3-5)	ACCZ	61	1	>0	
Δz	Eq. (3.3-6)	AXHI	61	8	>0	
D_h	Eq. (3.3-9)	DHZ	61	32	>0	
d_{sti}	Eq. (3.3-46)	DSTIZ	61	39	>0	
d_{sto}	Eq. (3.3-48)	DSTOZ	61	46	>0	
		PLENL	61	53	>0	
		RBR	61	54	> ROUTFP	
		RER	61	78	> RBR	
r_{brp}	Eq. (3.3-104)	RBRPL	61	102	>0	
r_{crp}	Eq. (3.3-109)	RERPL	61	103	> RBRPL	
		RINFP	61	104		
		ROUTFP	61	128	> RINFP	
		ZONEL	61	152	>0	
S_{pt}	Eq. (3.3-41)	SRFSTZ	61	159		
		AREAPC	61	166	>0	
d_{ro}	Eq. (3.3-80)	DRFO	61	169	>0	
		RBR0	61	180	>0	
		RER0	61	181	>0	
S_{cr}	Eq. (3.3-83)	SER	61	182	>0	
d_{ri}	Eq. (3.3-77)	DRFI	61	189	>0	
N_{PI}	Eq. (3.11-1)	XXNPIN	61	274	>0	
γ_s	Eq. (3.3-23)	GAMSS	62	2		

continues on next page

Table 3.15.1 – continued from previous page

Variable	Reference	Input Variable	Input Block	Location Number	Suggested Value	External Reference
γ_c	Eq. (3.3-6)	**GAMTNC**	62	4		
γ_e	Eq. (3.3-24)	**GAMTNE**	62	5		
		PSHAPE	62	6		
P_r	Eq. (3.3-23)	**PSHAPR**	62	30		
		AHBPAR	63	2		
		BHBPAR	63	3		
		CHBPAR	63	4		
		HBMAX	63	5		
		HBMIN	63	6		
		HBPAR	63	7		
k_{si}	Eq. (3.3-44)	**XKSTIZ**	63	11		
k_{so}	Eq. (3.3-47)	**XKSTOZ**	63	18		
$\varepsilon\sigma$	Eq. (3.3-4)	**DEL**	63	25		
k_r	Eq. (3.3-78)	**XKRF**	63	28		
$(\varrho c)_{sti}$	Eq. (3.3-46)	**RHOCSI**	63	37	>0	
$(\varrho c)_{sto}$	Eq. (3.3-48)	**RHOCSO**	63	44	>0	
$(\varrho c)_r$	Eq. (3.3-77)	**RHOCR**	63	51	>0	
$(\varrho c)_g$	Eq. (3.3-104)	**RHOCG**	63	58	>0	
R_g	Eq. (3.3-107)	**RG**	63	59	>0	
α_f	Eq. (3.10-4)	**FUELEX**	63	73		
α_e	Eq. (3.10-5)	**CLADEX**	63	74		
Y_f	Eq. (3.10-8)	**YFUEL**	63	75		
Y_e	Eq. (3.10-9)	**YCLAD**	63	76		
		FULREX	63	77		
		CLDREX	63	78		
$(HA)_{I,M}$	Eq. (3.11-1)	**HACHCH**	63	82		
$1/\tau_{ht}$	Eq. (3.3-12)	**TAUINV**	63	104	.5-5	
A_{fr}	Eq. (3.9-4), Eq. (3.9-5)	**AFR**	64	1	.1875	
b_{fr}	Eq. (3.9-4), Eq. (3.9-5)	**BFR**	64	2	-.2	
c_1	Eq. (3.3-9)	**C1**	64	3	.025	2-5
c_2	Eq. (3.3-9)	**C2**	64	4	.8	2-5
c_3	Eq. (3.3-9)	**C3**	64	5	4.8	2-5
		DWMAX	64	6	.2	
Re_L	Eq. (3.9-4), Eq. (3.9-5)	**RELAM**	64	7	2100	
A_{fL}	Eq. (3.9-4), Eq. (3.9-5)	**AFLAM**	64	8	64	
Re_{TR}		**RETRAN**	64	8		
C_1		**C1TRAN**	64	9		
C_2		**C2TRAN**	64	9		
C_3		**C3TRAN**	64	9		
		W0	64	47		
K_{or}	Eq. (3.9-17)	**XKORI**	64	48		
		XKORGD	64	64		

continues on next page

Table 3.15.1 – continued from previous page

Variable	Reference	Input Variable	Input Block	Location Number	Suggested Value	External Reference
$\left(\frac{\Delta z_i}{A}\right)_b$	Eq. (3.9-9)	**DZIAB**	64	65	>0	
$\left(\frac{\Delta z_i}{A}\right)_f$	Eq. (3.9-9)	**DZIAT**	64	66		
T_{out}	Eq. (3.3-151)	**DTLMAX**	64	69	15	
U_1	Eq. (3.10-1)	**TUPL**	64	74		
U_2	Eq. (3.10-2)	**UACH1**	64	189		
		UACH2	64	190		

3.15.1.3 Empty Reflector Region

Sometimes a reflector zone is used to represent an empty section of a subassembly. The duct wall is usually represented by the structure; and since there is nothing but sodium inside the duct wall, there is nothing for the reflector to represent. The code requires a "reflector" in every axial zone except the pin section, so some input for the reflector must be included. The reflector perimeter, SER, cannot be zero; but it can be set to a very small value, such as 10^{-9}, so that it would have no impact on coolant temperatures. Also, the total reflector thickness, DRFI plus DRFO, must be a reasonable value, 0.003 m or more, to prevent numerical instabilities in the reflector temperature calculations in the boiling module. The pre-voiding reflector temperature calculations are numerically stable for any reflector greater than zero.

3.15.1.4 Structure and Reflector Node Thickness

Two radial nodes are used in the reflectors and in the structure. The only restrictions on node thicknesses are that the total reflector thickness and the total structure thickness must be reasonable; 0.003 m or more, to prevent numerical instabilities in the temperature calculations in the boiling module. Usually, the node in contact with the coolant, the inner structure node or the outer reflector node, represents approximately 10% of the total thickness; and the other node represents the rest. Then the small node in contact the coolant will react rapidly to rapid changes in the coolant temperatures, whereas the larger node will dominate the longer-term response to the slow changes.

3.15.1.5 Coolant Re-entry Temperature

If PRIMAR-1 is being used, then the re-entry temperature calculation described in Section 3.3.6 uses the input variable TUPL for the bulk temperature of the coolant in the outlet plenum. If PRIMAR-4 is being used, then the outlet plenum temperature is computed by PRIMAR-4; and TUPL is not used.

3.15.2 Output Description

Figure 3.14.1 shows a typical thermal hydraulic output for one step of the transient. The output is largely self-explanatory. Note that on the first page of this output the axial location, coolant temperature, saturation temperature, and pressure are the values at node boundaries; whereas the cladding, structure, plenum, and reflector temperatures are the values at node mid-points. Also, on the second page the radial fuel temperatures are the values at node mid-points except for the inner and outer fuel temperature, as indicated in Figure 3.2.4; whereas the radii are the values of radial node boundaries.

CHANNEL 1 SAS4A 0.0 PRETEST ANALYSIS OF THE THORS NATURAL CIRCULATION TESTS 11/28/83 19.03.36 PAGE 69
PRIMAR4 OPTION, ONE CHANNEL

*** TRANSIENT STATE RESULTS ON TIME STEP 50 AT TIME 25.000 (SEC) IN CHANNEL 1 ***

FLOWRATE (KG/S) 0.0081239

COOLANT PROFILE

AXIAL LOCATION (M)	COOLANT TEMPERATURE (K)	SATURATION TEMPERATURE (K)	PRESSURE (MPA)	CLADDING TEMPERATURE (K)	STRUCTURE TEMPERATURE (K)		PLENUM TEMPERATURE (K)	REFLECTOR TEMPERATURE (K)	
					INNER	OUTER		INNER	OUTER
VESSEL OUTLET			0.1443						
3.36974	1200.624	1199.702	0.1443						
3.18559	1200.624	1200.795	0.1456		1200.624	1200.624		1200.624	1200.624
3.00144	1200.624	1201.830	0.1470		1200.624	1200.624		1200.624	1200.624
2.81729	1200.624	1202.958	0.1483		1200.624	1200.624		1200.624	1200.624
2.63314	1200.624	1204.028	0.1496		1200.624	1200.624		1200.624	1200.624
2.49344	1200.623	1204.829	0.1506		1200.624	1200.624		1200.624	1200.624
2.35374	1200.621	1205.627	0.1516		1200.623	1200.623		1200.623	1200.623
2.35174	1200.621	1205.634	0.1517		1200.622	1200.624		1200.621	1200.621
2.25064	1200.620	1206.350	0.1525		1200.621	1200.623		1200.620	1200.620
2.14954	1200.642	1207.013	0.1534		1200.625	1200.623		1200.628	1200.629
2.04844	1200.735	1207.673	0.1542		1200.658	1200.630		1200.676	1200.682
1.94734	1200.988	1208.330	0.1550		1200.770	1200.667		1200.830	1200.846
1.86267	1201.412	1208.877	0.1557	1201.175	1201.004	1200.761			
1.77801	1202.120	1209.423	0.1564	1201.726	1201.419	1200.955			
1.69334	1203.196	1209.966	0.1571	1202.601	1202.104	1201.319			
1.60867	1204.706	1210.507	0.1578	1203.874	1203.138	1201.932			
1.52401	1206.684	1211.046	0.1585	1205.598	1204.586	1202.878			
1.43934	1209.121	1211.583	0.1592	1207.783	1206.486	1204.230			
1.35467	1211.963	1212.118	0.1599	1210.413	1208.834	1206.033			
1.27000	1215.109	1212.650	0.1606	1213.399	1211.586	1208.315			
1.18534	1218.427	1213.180	0.1613	1216.629	1214.653	1211.027			
1.10067	1191.641	1213.711	0.1620	1208.505	1202.851	1199.039			
1.01600	1154.046	1214.245	0.1628	1177.533	1170.685	1166.855			
0.93134	1107.039	1214.784	0.1635	1136.220	1128.487	1124.780			
0.84667	1052.422	1215.327	0.1642	1086.130	1077.852	1074.402			
0.76200	992.352	1215.876	0.1649	1029.223	1020.745	1017.673			
0.67734	929.241	1216.432	0.1657	967.782	959.434	956.831			
0.59267	865.710	1216.994	0.1664	904.330	896.415	894.343			
0.50800	804.314	1217.563	0.1672	841.437	834.251	832.728			
0.42334	747.588	1218.137	0.1680	781.818	775.460	774.449			
0.33867	697.872	1218.716	0.1688	727.789	722.458	721.880			
0.25400	657.181	1219.299	0.1696	681.615	677.410	677.152			
0.16933	627.150	1219.885	0.1704	645.155	642.137	642.072			
0.08467	627.150	1220.469	0.1712	627.150	627.150	627.150			
0.0	627.150	1221.052	0.1720	627.150	627.150	627.150			
-0.08467	627.150	1221.632	0.1728	627.150	627.150	627.150	627.150		
-0.18627	627.150	1222.624	0.1742		627.150	627.150		627.150	627.150
VESSEL INLET			0.1742						

Figure 3.15.1: Sample Subassembly Thermal Hydraulics Output

CHANNEL 1 SAS4A 0.0 PRETEST ANALYSIS OF THE THORS NATURAL CIRCULATION TESTS 11/28/83 19.03.36 PAGE 70
PRIMAR4 OPTION, ONE CHANNEL

AXIAL LOCATION (M)	RADIAL FUEL TEMPERATURE MESH (K) 1	2	3	4	5	6	7	8
1.90501	1201.105	1201.106	1201.110	1201.116	1201.124	1201.135	1201.148	1201.164
1.82034	1201.613	1201.615	1201.621	1201.631	1201.644	1201.662	1201.683	1201.709
1.73567	1202.436	1202.439	1202.447	1202.462	1202.481	1202.507	1202.538	1202.576
1.65101	1203.652	1203.656	1203.668	1203.687	1203.714	1203.748	1203.791	1203.841
1.56634	1205.319	1205.324	1205.338	1205.363	1205.396	1205.440	1205.493	1205.556
1.48167	1207.457	1207.463	1207.480	1207.509	1207.549	1207.600	1207.663	1207.738
1.39701	1210.043	1210.049	1210.068	1210.101	1210.145	1210.203	1210.274	1210.357
1.31234	1213.005	1213.011	1213.032	1213.066	1213.114	1213.175	1213.251	1213.339
1.22767	1216.229	1216.236	1216.257	1216.292	1216.340	1216.403	1216.479	1216.569
1.14300	1524.195	1524.387	1524.961	1525.910	1527.225	1399.673	1296.009	1210.246
1.05834	1607.445	1607.656	1608.281	1609.312	1610.729	1437.373	1296.267	1179.911
0.97367	1662.236	1662.456	1663.107	1664.175	1665.636	1453.214	1280.635	1139.143
0.88900	1685.980	1686.199	1686.846	1687.903	1689.340	1446.133	1249.494	1089.483
0.80434	1677.735	1677.942	1678.555	1679.552	1680.898	1416.406	1204.114	1032.876
0.71967	1638.378	1638.565	1639.116	1640.009	1641.206	1365.719	1146.624	971.592
0.63500	1569.130	1569.307	1569.829	1570.670	1571.792	1296.382	1079.663	908.146
0.55034	1468.693	1468.831	1469.236	1469.887	1470.747	1209.793	1006.856	845.162
0.46567	1348.134	1348.232	1348.520	1348.980	1349.582	1112.997	931.261	785.207
0.38100	1210.525	1210.586	1210.767	1211.054	1211.423	1010.987	856.297	730.758
0.29633	1065.361	1065.390	1065.475	1065.610	1065.783	908.438	785.147	684.045
0.21167	921.284	921.295	921.326	921.375	921.438	809.560	720.587	646.950
0.12700	627.150	627.150	627.150	627.150	627.150	627.150	627.150	627.150
0.04233	627.150	627.150	627.150	627.150	627.150	627.150	627.150	627.150

FUEL NODE	RADIAL FUEL MESH (CM) 1	2	3	4	5	6	7	8	9	IZ	IETA
23	0.0	0.0222	0.0667	0.1111	0.1556	0.2000	0.2445	0.2889	0.3112	0	0
22	0.0	0.0222	0.0667	0.1111	0.1556	0.2000	0.2445	0.2889	0.3112	0	0
21	0.0	0.0222	0.0667	0.1111	0.1556	0.2000	0.2445	0.2889	0.3112	0	0
20	0.0001	0.0223	0.0668	0.1112	0.1556	0.2001	0.2445	0.2889	0.3112	0	0
19	0.0001	0.0223	0.0668	0.1112	0.1556	0.2001	0.2445	0.2889	0.3112	0	0
18	0.0001	0.0223	0.0668	0.1112	0.1556	0.2001	0.2445	0.2889	0.3112	0	0
17	0.0001	0.0223	0.0668	0.1112	0.1556	0.2001	0.2445	0.2889	0.3112	0	0
16	0.0001	0.0223	0.0668	0.1112	0.1556	0.2001	0.2445	0.2889	0.3112	0	0
15	0.0001	0.0223	0.0668	0.1112	0.1556	0.2001	0.2445	0.2889	0.3112	0	0
14	0.0001	0.0223	0.0668	0.1112	0.1556	0.2001	0.2445	0.2889	0.3112	0	0
13	0.0001	0.0223	0.0668	0.1112	0.1556	0.2001	0.2445	0.2889	0.3112	0	0
12	0.0001	0.0223	0.0668	0.1112	0.1556	0.2001	0.2445	0.2889	0.3112	0	0
11	0.0001	0.0223	0.0668	0.1112	0.1556	0.2001	0.2445	0.2889	0.3112	0	0
10	0.0001	0.0223	0.0668	0.1112	0.1556	0.2001	0.2445	0.2889	0.3112	0	0
9	0.0001	0.0223	0.0668	0.1112	0.1556	0.2001	0.2445	0.2889	0.3112	0	0
8	0.0001	0.0223	0.0668	0.1112	0.1556	0.2001	0.2445	0.2889	0.3112	0	0
7	0.0001	0.0223	0.0668	0.1112	0.1556	0.2001	0.2445	0.2889	0.3112	0	0
6	0.0001	0.0223	0.0668	0.1112	0.1556	0.2001	0.2445	0.2889	0.3112	0	0
5	0.0001	0.0223	0.0668	0.1112	0.1556	0.2001	0.2445	0.2889	0.3112	0	0
4	0.0001	0.0223	0.0668	0.1112	0.1556	0.2001	0.2445	0.2889	0.3112	0	0
3	0.0001	0.0223	0.0668	0.1112	0.1556	0.2001	0.2445	0.2889	0.3112	0	0
2	0.0001	0.0223	0.0668	0.1112	0.1556	0.2001	0.2445	0.2889	0.3112	0	0
1	0.0001	0.0223	0.0668	0.1112	0.1556	0.2001	0.2445	0.2889	0.3112	0	0

Figure 3.15.2: Sample Subassembly Thermal Hydraulics Output (Cont'd.)

CHANNEL 1 SAS4A 0.0 PRETEST ANALYSIS OF THE THORS NATURAL CIRCULATION TESTS 11/28/83 19.03.36 PAGE 71
PRIMAR4 OPTION, ONE CHANNEL

AXIAL LOCATION (M)	CLADDING TEMPERATURE (K)			AVERAGE COOLANT TEMPERATURE (K)	STRUCTURE TEMPERATURE (K)		AVERAGE FUEL TEMPERATURE (K)	RADIAL LOCATION (CM)			
	INNER	MIDPOINT	OUTER		INNER	OUTER		FUEL INNER	FUEL OUTER	CLAD INNER	CLAD OUTER
1.90501	1201.165	1201.175	1201.186	1201.200	1201.004	1200.761	1201.135	0.0	0.3112	0.3112	0.3492
1.82034	1201.709	1201.726	1201.744	1201.766	1201.419	1200.955	1201.662	0.0	0.3112	0.3112	0.3492
1.73567	1202.576	1202.601	1202.627	1202.658	1202.104	1201.319	1202.506	0.0	0.3112	0.3112	0.3492
1.65101	1203.841	1203.874	1203.909	1203.951	1203.138	1201.932	1203.747	0.0001	0.3112	0.3112	0.3492
1.56634	1205.557	1205.598	1205.642	1205.695	1204.586	1202.878	1205.439	0.0001	0.3112	0.3112	0.3492
1.48167	1207.738	1207.788	1207.840	1207.902	1206.486	1204.230	1207.599	0.0001	0.3112	0.3112	0.3492
1.39701	1210.358	1210.413	1210.471	1210.542	1208.834	1206.038	1210.202	0.0001	0.3112	0.3112	0.3492
1.31234	1213.340	1213.399	1213.461	1213.536	1211.536	1208.315	1213.174	0.0001	0.3112	0.3112	0.3492
1.22767	1216.569	1216.629	1216.692	1216.768	1214.653	1211.027	1216.401	0.0001	0.3112	0.3112	0.3492
1.14300	1210.227	1208.505	1206.885	1205.034	1202.851	1199.039	1400.460	0.0001	0.3112	0.3112	0.3492
1.05834	1179.836	1177.533	1175.318	1172.843	1170.685	1166.855	1438.522	0.0001	0.3112	0.3112	0.3492
0.97367	1139.112	1136.220	1133.496	1130.542	1128.487	1124.780	1454.994	0.0001	0.3112	0.3112	0.3492
0.88900	1089.448	1086.130	1083.003	1079.731	1077.852	1074.402	1448.894	0.0001	0.3112	0.3112	0.3492
0.80434	1032.837	1029.223	1025.815	1022.387	1020.745	1017.673	1420.434	0.0001	0.3112	0.3112	0.3492
0.71967	971.552	967.782	964.226	960.796	959.434	956.831	1371.251	0.0001	0.3112	0.3112	0.3492
0.63500	908.106	904.330	900.768	897.475	896.415	894.343	1303.302	0.0001	0.3112	0.3112	0.3492
0.55034	845.123	841.487	838.056	835.012	834.251	832.728	1217.457	0.0001	0.3112	0.3112	0.3492
0.46567	785.171	781.818	778.653	775.951	775.460	774.449	1120.946	0.0001	0.3112	0.3112	0.3492
0.38100	730.726	727.789	725.017	722.730	722.458	721.880	1017.249	0.0001	0.3112	0.3112	0.3492
0.29633	684.020	681.615	679.345	677.526	677.410	677.152	912.334	0.0001	0.3112	0.3112	0.3492
0.21167	646.931	645.155	643.479	642.165	642.137	642.072	811.611	0.0001	0.3112	0.3112	0.3492
0.12700	627.150	627.150	627.150	627.150	627.150	627.150	627.150	0.0001	0.3112	0.3112	0.3492
0.04233	627.150	627.150	627.150	627.150	627.150	627.150	627.150	0.0001	0.3112	0.3112	0.3492

FUEL NODE	AXIAL LOCATION (M)	FRACTION MOLTEN	ABOVE SOLDUS	ABOVE LIQDUS	GAP CONDUCTANCE W/M2-K	
23	1.90501	0.0	0.0	0.0	1.0000D+07	0.0
22	1.82034	0.0	0.0	0.0	1.0000D+07	0.0
21	1.73567	0.0	0.0	0.0	1.0000D+07	0.0
20	1.65101	0.0	0.0	0.0	1.0000D+07	0.0
19	1.56634	0.0	0.0	0.0	1.0000D+07	0.0
18	1.48167	0.0	0.0	0.0	1.0000D+07	0.0
17	1.39701	0.0	0.0	0.0	1.0000D+07	0.0
16	1.31234	0.0	0.0	0.0	1.0000D+07	0.0
15	1.22767	0.0	0.0	0.0	1.0000D+07	0.0
14	1.14300	0.0	0.0	0.0	1.0000D+07	0.0
13	1.05834	0.0	0.0	0.0	1.0000D+07	0.0
12	0.97367	0.0	0.0	0.0	1.0000D+07	0.0
11	0.88900	0.0	0.0	0.0	1.0000D+07	0.0
10	0.80434	0.0	0.0	0.0	1.0000D+07	0.0
9	0.71967	0.0	0.0	0.0	1.0000D+07	0.0
8	0.63500	0.0	0.0	0.0	1.0000D+07	0.0
7	0.55034	0.0	0.0	0.0	1.0000D+07	0.0
6	0.46567	0.0	0.0	0.0	1.0000D+07	0.0
5	0.38100	0.0	0.0	0.0	1.0000D+07	0.0
4	0.29633	0.0	0.0	0.0	1.0000D+07	0.0
3	0.21167	0.0	0.0	0.0	1.0000D+07	0.0
2	0.12700	0.0	0.0	0.0	1.0000D+07	0.0
1	0.04233	0.0	0.0	0.0	1.0000D+07	0.0

Figure 3.15.3: Sample Subassembly Thermal Hydraulics Output (Cont'd.)

3.15.3 Subchannel Information

This section provides additional information that is necessary to implement the subchannel model described in Section 3.14. This includes a brief user guide, followed by a sample problem, then a detailed overview of the required input. More information about the SAS4A/SASSYS-1 preprocessor, which is available to expedite the input creation process for the subchannel model, can be found in Section 3.19.2.

3.15.3.1 User Guide

This user guide provides a quick summary of the grouping of channels and subassemblies and how to properly prepare the required input. Information about standard output and a binary plotting option is also given.

3.15.3.1.1 Channels, Subassemblies, and Groups

A basic feature of the core treatment is the concept of a channel. In simple terms a channel consists of a fuel pin and its associated coolant and structure. As used in the SAS4A/SASSYS-1 code, a channel is a bit more involved. Figure 3.15.4 shows some details of a SAS4A/SASSYS-1 channel if the three dimensional thermal hydraulics model is *not* being used. In the axial direction the whole length of the subassembly is modeled. A channel includes the whole fuel pin, including the fuel region, axial reflectors in the fuel pin, a gas plenum region at the top or bottom of the fuel pin. Also, a channel includes axial reflector zones above and below the fuel pins.

When the three dimensional thermal hydraulics model is used, the concept of a channel is even more involved. The three-dimensional treatment is used only in the pin section. Multiple channels are used in the pin section of a subassembly, but in the axial reflector zones a single channel treatment is used, as shown in Figure 3.15.5. The first channel used to represent a subassembly includes the pin section and the axial reflectors. All other channels used to represent the subassembly only include the pin section, as seen in Figure 3.15.6.

Coolant flow enters a single channel at the bottom of the subassembly, flows through a single channel in the lower reflectors, splits into multiple flow paths in the pin section, and re-combines into a single flow path for the upper reflectors, as shown in Figure 3.15.7. In the pin section, cross-flow between the channels representing a subassembly is accounted for.

If one is modeling a reactor without solid duct walls, then accounting for cross-flow between subassemblies is desirable. In order to handle this situation, the concept of a group of channels was incorporated in the model. If solid duct walls are used, then there will be a one-to-one correspondence between subassemblies and groups. If solid duct walls are not used then a group can include many subassemblies, maybe even the whole core. Cross flow between adjacent channels in the whole group is accounted for, but again cross flow is only accounted for in the pin section. The reflector zones are included only in the first channel in a group. Channels are still combined into subassemblies for some edits, even if multiple subassemblies are included in a group. The maximum number of channels that can be included in a group is limited only by the amount of computer memory available for temporary storage of coolant variables during the simultaneous solution of the coolant conditions in all channels in the group. This temporary storage memory is allocated dynamically at run time based on the maximum group size.

Subassembly numbers are assigned by the user by setting the input variable ISUBAS (location 89 of block 51). Group numbers are assigned by the code on the basis of input variable JJMLTP (location 214 of block 51). If JJMLTP for a channel is greater than or equal to zero, then a new group is started. If JJMLTP < 0, then the channel is part of the current group.

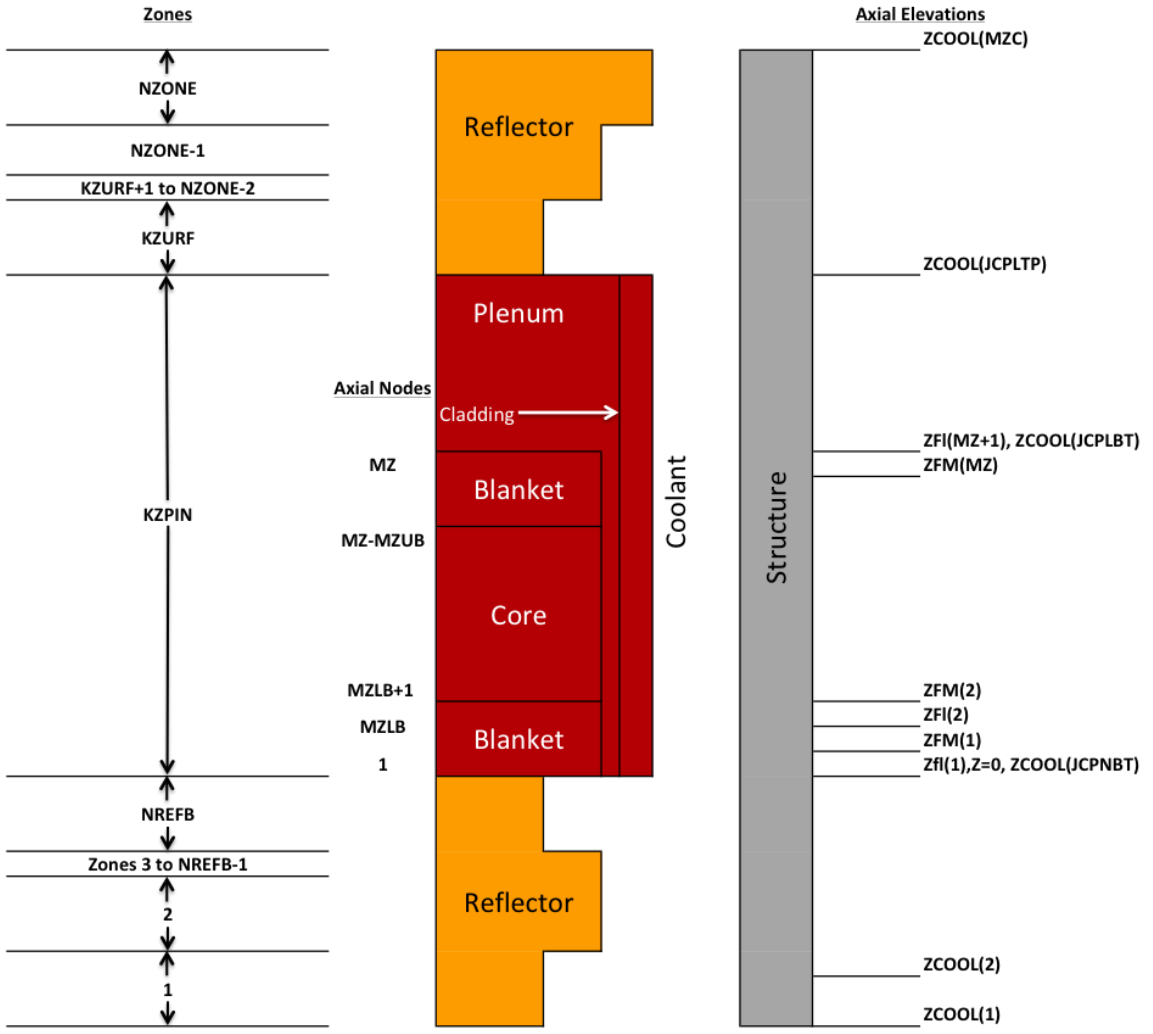


Figure 3.15.4: Details of SAS4A/SASSYS-1 Axial and Radial Nodes (if 3-D Thermal Hydraulics Model is not Being Used)

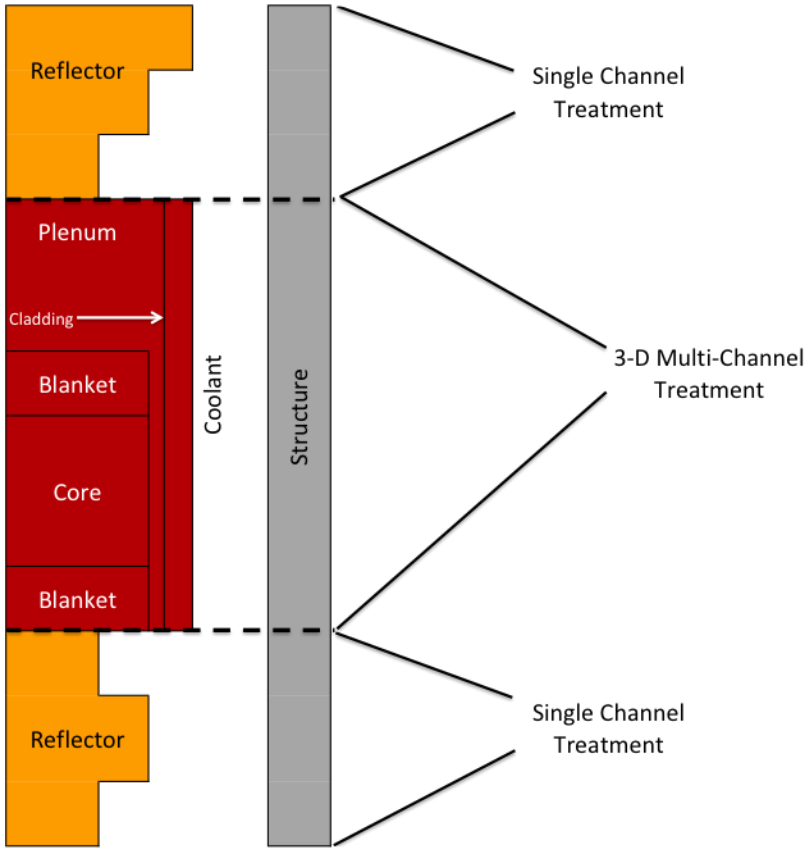


Figure 3.15.5: Breakdown of Axial Regions and Treatments

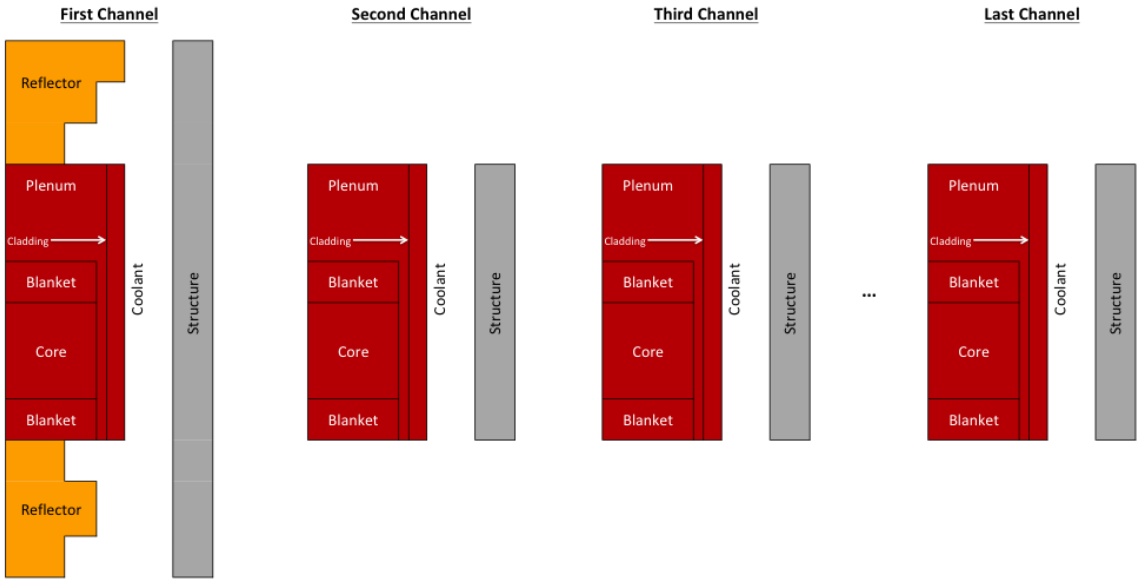


Figure 3.15.6: 3-D Thermal Hydraulics Input Channel Structure

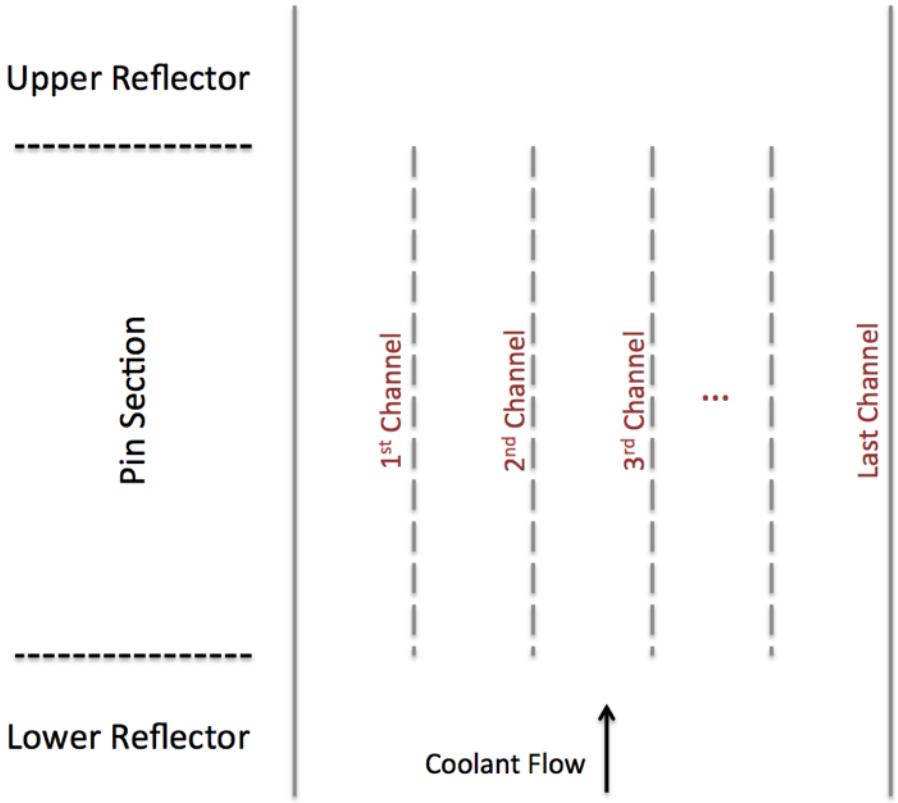


Figure 3.15.7: Subchannel Model Flow Diagram

3.15.3.1.2 Preparing Input for a Case

One issue to be addressed when preparing input for a case using the new three dimensional thermal hydraulics core model is the amount of detail to be used for various parts of the core. If enough computer memory and computer speed are available, then one could analyze every coolant subchannel and every fuel pin in the core. It is also possible to analyze some subassemblies with the detailed subchannel treatment shown in Figure 3.14.1, to analyze other subassemblies with a somewhat less detailed treatment such as that shown in Figure 3.15.8, and to analyze the rest of the core using one channel to represent a subassembly or a group of similar subassemblies.

The amount of input required for even a moderately sized case can be considerable. Some input is required for each channel, and the number of channels can easily be very large. Fortunately, most of the input for the new model is fairly repetitive, and much of it can be automated by a preprocessor. Section 3.19.2.3 describes a preprocessor that has been written for the SAS4A/SASSYS-1 code with the new model.

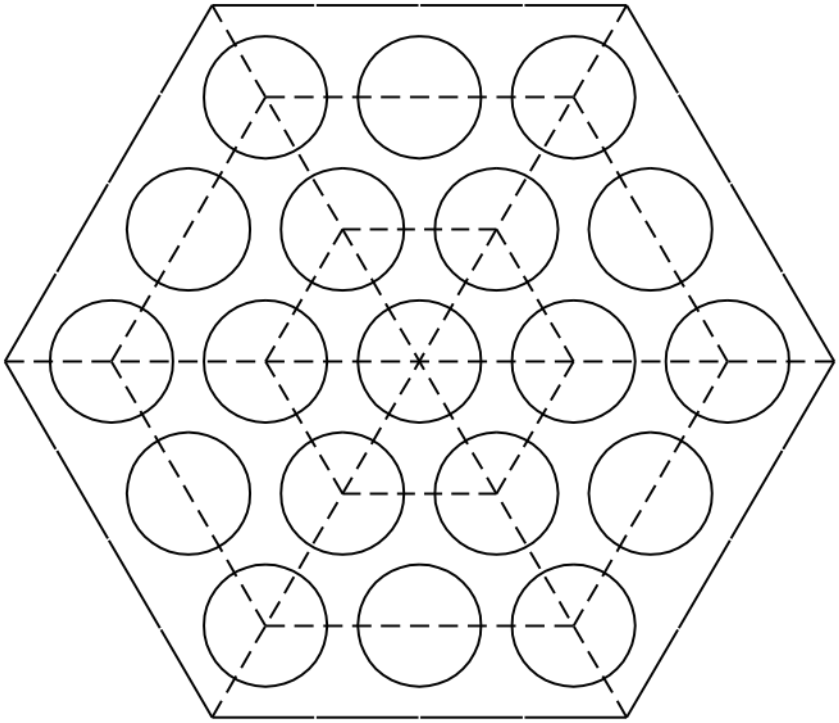


Figure 3.15.8: Subchannels Grouped in Rings

3.15.3.1.3 Controlling the Amount of Standard Output

The standard output from SAS4A/SASSYS-1 includes tables of temperatures, pressures and coolant flow rates for each axial node and each radial node in each channel. These tables are printed out for the steady-state results and for every IPO (location 12, input block 1) time steps during the transient. Even before the new three-dimensional thermal hydraulic model was added to the code, the problem of excessive amounts of output from the code was recognized. An input variable, IPRSKP (location 274, block 51) can be used to skip output from specified channels. Also, before the new model was added to the code, reactivity feedback components for each channel were printed out for each step of the transient. Now the reactivity feedback components are printed out for each subassembly rather than for each channel.

3.15.3.1.4 Binary Output for Plotting

The new model includes a provision for writing a binary output file on unit 24. This output file can contain user specified values calculated by the new model. It is anticipated that plotting programs will use this file to plot results from the new model. The user can specify which data is included on unit 24, how often that data is written, and for which channels.

The SAS4A/SASSYS-1 code writes a number of binary output files other than the unit 24 file mentioned above. The CHANNEL.dat binary file reports reactor power and reactivity feedback in addition to temperatures of fuel, clad, and structure, and coolant flow rates for all channels. The CHANNEL.dat file can be extremely large if using the subchannel model, as a full thermal hydraulic dataset is provided for each subchannel in the binary output. While the user cannot control what channels are included in CHANNEL.dat, the user can control the frequency of output using the MSTPLA, MSTPLB, MSTPL1, MSTPL2, and MSTPL3 (INPCOM:107-111) input fields.

3.15.3.2 Sample Problem

Figure 3.15.9 shows output from a sample problem run with the new model. This test case uses two 19 pin subassemblies with subassembly-to-subassembly heat transfer. Selected coolant temperatures from one of the subassemblies are shown in Figure 3.15.9. The transient was a simulated scram with a delayed pump trip and little natural circulation head in the primary loop. Temperatures drop rapidly due to the scram and then rise after the pump trip until enough buoyancy is established in the subassemblies to increase the flow rate and stop the temperature rise. Flow reversal occurred in the center subchannels from 22.5 seconds to 83.5 seconds in the transient. Flow reversal occurred in the edge subchannels from 22.0 seconds to 111.25 seconds.

The case described above used two subassemblies, each containing 19 pins and 42 coolant subchannels in the pin section. Four axial nodes were used in a lower reflector below the pin section, 24 axial nodes were used in the core, 6 axial nodes were used in the gas plenum region above the core, and 5 axial nodes were used in a reflector above the pins. A null transient of 50 steps with a time step of 1.0 second was used for each subassembly separately. Then a 50 step null transient using both subassemblies and subassembly-to-subassembly heat transfer was run with a time step of 0.5 seconds. Finally the regular transient was run for 1200 steps using a time step of 0.25 seconds. The null transients started with 35 iterations per step and got down to 2 iterations per step as they converged on a steady-state solution. The regular transient required between 12 and 21 iterations per step. The total running time was 867 seconds on a Sun Blade computer with a 500 MHz processor.

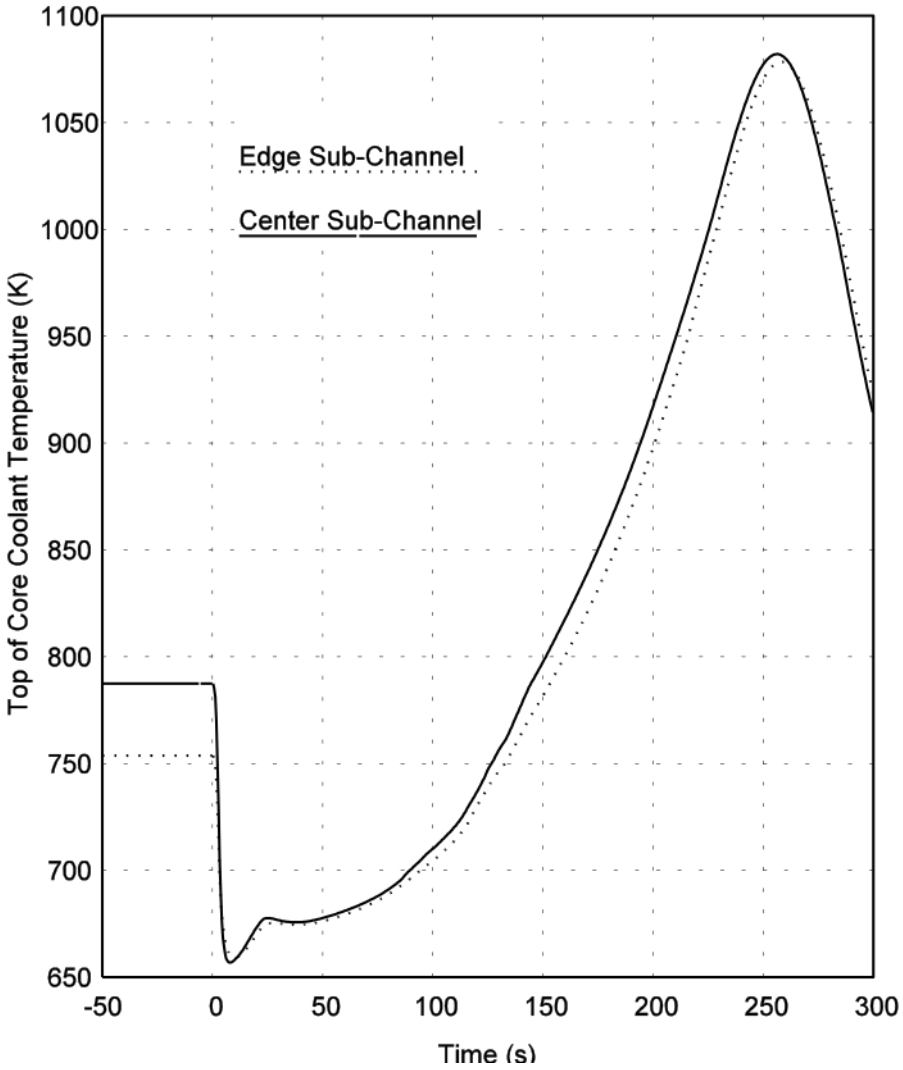


Figure 3.15.9: Results for a Scram with a Delayed Pump Trip

3.15.3.3 Input Description

Input for the SAS4A/SASSYS-1 code is read into numbered locations in input blocks. Each input block has a name and a number. Below is a listing of the additional input variables that are used for the new three-dimensional thermal-hydraulic core model. A description of each input can be found in the User Guide in Chapter 2.

Table 3.15.2: Detailed Subchannel Model Input Variables

Variable	Reference Eq. No.	Input Variable	Input Block	Location Number	Suggested Value	External Reference
		ISSNUL	1	87		
		ISCH	1	115		
		ISUBAS	51	89		
		JJMLTP	51	214		
		JCHMPN(K)	51	215-218		
		NUMKLT	51	219		
		KSWIRL	51	220		
		NULSTI	51	221		
		IFT24	51	495		
		ILATF	51	496		
		NFT24	51	501		
		JCFT24(K)	51	502-521		
		ITYPE24(K)	51	522-541		
		NULPT1	51	542		
	Eq. (3.14-7)	UACHM1(K)	64	197-200		
	Eq. (3.14-8)	UACHM2(K)	64	201-204		
		ALATRL(K)	64	205-208		
		XKLAT(K)	64	209-212		
		DPLTLM	64	213		
		XKSWRL	64	214		
		DTNUL1	64	215		
		EPSFLW	64	216		
		EPSTMP	64	217		
		EPSPRS	64	218	1.0	
		XCMPRS	64	219		
		XLINRT	64	220		

3.16 Thermal Properties of Fuel and Cladding

All material property data for fuel and cladding are cast as functions or subroutines to allow for modularization and the ease of making changes. This also allows for the incorporation of different materials data in a straightforward manner. In a number of the correlations used, the units are inconsistent with the SI unit system adopted by SAS4A/SASSYS-1. The routines that use these correlations carry out the appropriate units conversions internally.

The thermal properties for fuel and cladding are described in this section. Sodium properties are described in Chapter 12.

3.16.1 Fuel Density

The fuel density can be obtained either from a user-supplied table of density vs temperature or from a correlation with

$$\rho_{\mathrm{f}} = \frac{\rho_{\mathrm{o}}}{1 + C_1 (T - 273) + C_2 (T - 273)^2} \tag{3.16-1}$$

where

ρ_{o} = The theoretical density at 273 K, kg/m^3

C_1, C_2 = Input coefficients

T = Temperature, K

This applies between 273 K and the solidus temperature.

The liquid fuel density is given by

$$\rho_{\mathrm{l}} = \frac{\rho_{\mathrm{o}}}{1 + C_3 (T - 273)} \tag{3.16-2}$$

where

C_3 = Input coefficient

This applies to temperatures above the liquidus. For the range between the solidus and liquidus temperatures, a linear interpolation is performed.

These equations are found in the function RHOF. Suggested values of coefficients are from the Nuclear Systems Materials Handbook [3-13].

ρ_{o} = **COEFDS**(1) = 11.05×10^3kg/m^3 (mixed oxide)

C_1 = **COEFDS**(2) = 2.04×10^{-5} K^{-1}

C_2 = **COEFDS**(3) = 8.70×10^{-9} K^{-2}

C_3 = **COEFDL**(2) = 9.30×10^{-5} K^{-1}

3.16.2 Fuel Thermal Conductivity

Four different options exist for the fuel thermal conductivity. These are controlled through the input parameter **IRHOK**.

IRHOK = 0

The thermal conductivity as function of temperature is input in table form through the variable arrays XKTAB and XKTEM.

IRHOK = 1

For this option, the conductivity equations are given by:

$$k_1 (T) = 1.1 + \frac{1 \times 10^2}{T (.4888 - .4465 f_{\mathrm{D}})} \tag{3.16-3}$$

for $800°C \leq T \leq 2000°$ C

$$k_2 (T) = k_1 (800) \frac{168.844}{12.044 + (0.196) T} \tag{3.16-4}$$

for $T \leq 800^\circ$ C

$$k_3(T) = k_1(2000) \tag{3.16-5}$$

for $T > 2000^\circ$ C

where

k_1, k_2, k_3 = Fuel thermal conductivity, W/m-k

T = Temperature, °C

f_D = Fuel fraction of theoretical density

IRHOK = 2

This form of the conductivity is given by

$$k_1(T) = [(C_1 - f_D) f_D - 1] \left[\frac{1}{(C_2 + C_3 T)} + C_4 T^3 \right] \tag{3.16-6}$$

for $0.75 \leq f_D \leq 0.95$

$$k_2(T) = [3.0 f_D - 1] \left[\frac{1}{(C_5 + C_6 T)} + C_7 T^3 \right] \tag{3.16-7}$$

for $f_D > 0.95$

where

C_1, C_2, C_3, C_4, C_5, C_6, C_7 = Input variables

k_1, k_2 = Fuel conductivity W/m-k

T = Temperature, K

If T is greater than the melting temperature, it is set to the melting temperature.

Suggested values:

C_1 = **COEFK**(1) = 2.1

C_2 = **COEFK**(2) = 2.88×10^{-3}

C_3 = **COEFK**(3) = 2.52×10^{-5}

C_4 = **COEFK**(4) = 2.83×10^{-10}

C_5 = **COEFK**(5) = 5.75×10^{-2}

C_6 = **COEFK**(6) = 5.03×10^{-4}

C_7 = **COEFK**(7) = 2.91×10^{-11}

IRHOK = 3

This conductivity form is [3-14]

$$k_1(T) = \frac{4.005 \times 10^3}{(T - 273) + 402.4} + 0.6416 \times 10^{-10} T^3 \tag{3.16-8}$$

where

T = Temperature, K

k_1 = Conductivity in W/m-k

This is the correlation for UO_2 and is converted to mixed oxide by subtracting 0.2.

$$k_2(T) = k_1(T) - 0.2 \tag{3.16-9}$$

The porosity correction term was derived for use in the COMETHE-IIIJ [3-15] code and is given by

$$f_p = 1 - 1.029\varepsilon - 3.2\varepsilon^2 - 40.1\varepsilon^3 + 158\varepsilon^4 \tag{3.16-10}$$

where

f_p = Porosity multiplier

$\epsilon = 1 - \rho_f$ = Fractional porosity

ρ_f = fractional fuel density = actual density/theoretical density

The conductivity is therefore given by

$$k(T) = f_p k_2(T) \tag{3.16-11}$$

Two different routines contain the above correlations, FK and KFUEL. The function FK returns a single value of the conductivity for a single invocation and is used in the steady-state calculation. The subroutine KFUEL returns the conductivity values for each radial node in the current axial segment. It is used in the transient calculational procedure.

3.17 References

1. M. Lees, "An Extrapolated Crank-Nicholson Difference Scheme for Quasilinear Parabolic Equations" *Proc. Symp. On Nonlinear Partial Differential Equations*, W. F. Ames, Ed., Academic Press, 1966.
2. F. E. Dunn, et al., "The SAS2A LMFBR Accident Analysis Computer Code," ANL-8138, Argonne National Laboratory, 1974.
3. "A Preliminary User's Guide to Version 1.0 of the SAS3D LMFBR Accident Analysis Computer Code," J. E. Cahalan and D. R. Ferguson, Eds.; Available through the Reactor Analysis Division, Argonne National Laboratory, 1977.
4. W. M. Rohsenow and H. Y. Choi, *Heat, Mass and Momentum Transfer*, p. 189, Prentice-Hall, 1961.
5. W. M. Rohsenow and J. P. Hartnett, Eds., *Handbook of Heat Transfer*, p. 7-33, McGraw-Hill Book company, New York, 1973.
6. William T. Sha, Frank J. Goldner, Paull R. Heubotter, and Robert C. Schmitt, "Thermal Hydraulic Multichannel Analysis," *Proceedings of the International Meeting on Reactor Heat Transfer*, Karlsruhe, Germany, p. 180, October 9-11, 1973.
7. D. Mohr, L. Chang, and H. P. Planchon, "Validation of the HOTCHAN Code for Analyzing the EBR-II Core Following an Unprotected Loss of Flow," *Trans. Am. Nucl. Soc.*, vol. 57, p. 318, 1988.
8. Y. M. Kwon, Y. B. Lee, W. P. Chang and D. Hahn, "SSC-K Code User's Manual (Rev. 1)", KAERI/TR-2014, KAERI, Daejeon, Korea, 2002.
9. C. W. Stewart, C. L. Wheeler, R. J. Cena, C.A. McMonagle, J.M. Cuta and D.S. Trent, "COBRA-IV: the Model and the Method," BNWL-2214, Battelle Pacific Northwest Laboratories, Richland, Washington, 1977.
10. K. L. Basehore and N. E. Todreas, "SUPERENERGY-2: a MULTIASSEMBLY Steady-State Computer Code for lmfbr core Thermal-hydraulics Analysis," pnl-3379, pacific northwest laboratory, Hanford, Washington, 1980.

11. P. M. Magee, A. E. Dubberley, A. J. Lipps and T. Wu, "Safety Performance of the Advanced Liquid Metal Reactor," *Proceedings of ARS '94 International Topical Meeting on Advanced Reactors Safety,* Pittsburg, Pennsylvania, April 17-21, vol. 2, pp. 826-833, 1994.

12. S. K. CHENG and N. E. Todreas, "Hydrodynamic Models and Correlations for Bare and Wire-Wrapped Hexagonal Rod Bundles - Bundle Friction Factors, Subchannel Friction Factors and Mixing Parameters," *Nucl. Eng. And Design*, vol. 92, p. 227-251, 1986.

13. "Nuclear Systems Materials Handbook", TID-26666, Hanford Engineering Development Laboratory.

14. S. Y. Ogawa, E. A. Lees and M. F. Lyons, "Power Reactor High Performance UO_2 Program, Fuel Design Summary and Progress Status," GEAP-5591, General Electric Company, 1968.

15. P. Verbeek and H. Hoppe, "COMETHE-IIIJ: A Computer Code for Predicting Mechanical and Thermal Behavior of a Fuel Pin, Part 1: General Description," INBFR-107 (original report BN-7609-07), Belgonucleaire, Brussels, Belgium, 1977.

16. S.K. Chen, Y.M. Chen, and N.E. Todreas, "The upgraded Cheng and Todreas correlation for pressure drop in hexagonal wire-wrapped rod bundles," *Nucl. Eng. And Design*, vol. 335, p. 356-373, 2018.

3.18 Nomenclature

Subscript	Description
c	Coolant
e	Cladding
f	Fuel
g	Plenum gas
kz	Reflector zone
p	Gas plenum region
si	Structure inner node
so	Structure outer node
1	Beginning of the time step
2	End of the time step

Symbol	Description	Units
A_c	Coolant flow area	m^2
A_{ep}	Cross sectional area of the clad in the gas plenum region	m^2
A_g	Cross sectional area of the plenum gas	m^2
A_{fr}, b_{fr}	Coefficients in the friction factor correlation: $f = A_{fr}\,(\mathrm{Re})^{b_{fr}}$	–
c	Specific heat	J/kg-K
$\overline{c}_c$	Coolant specific heat	J/kg-K
c_e	Cladding specific heat	J/kg-K
$\overline{c}_f$	Average fuel specific heat, averaged over a time step	J/kg-K
c_m	Modified specific heat in the melting range	J/kg-K

continues on next page

Table 3.18.1 – continued from previous page

Symbol	Description	Units
c_{mix}	Coolant specific heat in the mixing zone used for re-entry temperature calculation	J/kg-K
c_1, c_2, c_3	Correlation constants used in coolant heat-transfer coefficients	–
D	Right-hand-side terms in the matrix equations for radial temperature profiles	J/m
D_h	Hydraulic diameter	m
d_{sti}	Structure inner node thickness	m
d_{sto}	Structure outer node thickness	m
E_{ec}	Heat flux from clad to coolant, integrated over a time step	J/m^2
E_{sc}	Heat flux form structure to coolant, integrated over a time step	J/m^2
f	Friction factor	–
f_i	Fraction of the structure thickness represented by the inner node	–
f_o	Fraction of the structure thickness represented by the outer node	–
g	Acceleration of gravity	m/s^2
h_b	Bond gap conductance	W/m^2-K
h_c	Coolant-film heat-transfer coefficient	W/m^2-K
h_{cond}	Condensation heat-transfer coefficient for sodium vapor	W/m^2-K
H_{eg}	Heat-transfer coefficient form the gas in the gas plenum to the cladding	W/m^2-K
H_{erc}	Heat-transfer coefficient from the cladding or reflector outer node to the coolant	W/m^2-K
h_r	Equivalent radiation heat-transfer coefficient	W/m^2-K
H_{rio}	Heat-transfer coefficient from the structure inner node to the reflector outer node	W/m^2-K
H_{sic}	Heat-transfer coefficient from the structure inner node to the coolant	W/m^2-K
H_{stio}	Heat-transfer coefficient from the structure inner node to the structure outer node	W/m^2-K
i	Radial node number	–
ic	Core channel number	–
I_1	Inertial integral in the momentum equation	m^{-1}
I_2	Acceleration integral in the momentum equation	m/kg
I_3	Friction integral in the momentum equation	m/kg
I_4	Orifice term in the momentum equation	m/kg
I_5	Density integral in the momentum equation	kg/m^2
j	Fuel axial node number	–
jc	Coolant axial node number	–
JC	Axial node number	–
k	Thermal conductivity	W/m-K
k_{ep}	Cladding thermal conductivity in the gas plenum region	W/m-K
$\overline{k}_{i\,j}$	Weighted average thermal conductivity for heat flow from node i to j	W/m-K
K_{or}	Orifice coefficient	–
L	1 for subassembly inlet, 2 for outlet	–
MZC	Total number of coolant axial nodes	–
m_e	Cladding mass	kg
m_f	Fuel mass	kg
M_{mix}	Mass of sodium in the mixing volume	kg
NC	Radial node number of the coolant node	–
NE	Radial node number for the cladding mid-point	–
NE′	Radial node number for the cladding outer surface node	–

continues on next page

Table 3.18.1 – continued from previous page

Symbol	Description	Units
NE″	Radial node number for the cladding inner surface node	–
NN	NT-1	–
N_{ps}	Number of fuel pins represented by a channel	–
NR	Radial node number for the fuel outer surface radius. (note NR = NE″)	–
NSI	Radial node number for the inner structure node	–
NSO	Radial node number for the outer structure node	–
NT	Radial node number for the fuel outer surface temperature node	–
p	Pressure	Pa
p_b	Pressure at the bottom of the subassembly	Pa
p_{b1}, p_{b2}	p_b at beginning and end of a time step	Pa
p_{in}	Pressure in the coolant inlet plenum	Pa
$\overline{P}_j$	Heat production rate in axial node j	W
P_r	Radial power shape, per unit mass	–
p_t	Pressure at the top of the subassembly	Pa
p_{t1}, p_{t2}	p_t at the beginning and end of a time step	Pa
p_x	Pressure in the coolant outlet plenum	Pa
$\left(\frac{\partial p}{\partial z}\right)_{fr}$	Friction pressure drop	Pa/m
$\left(\frac{\partial p}{\partial z}\right)_{k}$	Orifice pressure drop	Pa/m
Q_c	Coolant heat source due to direct heating by neutrons and gamma rays	W/m^3
Q_{ct}	Total steady-state heat source per unit of coolant volume	W/m^3
Q_{ec}	Heat flow from clad to coolant	W/m^3
q_{fe}	Fuel-to-cladding heat flux	W/m^2
Q_{sc}	Heat flow from structure to coolant	W/m^3
$Q_{sm}(i)$	Sum of the heat sources for all radial nodes inside and including node i	W
Q_{st}	Structure heat source due to direct heating by neutrons and gamma rays	W/m^2
Q_v	Heat source per unit volume	W/m^3
r	Radius	m
r_{brp}	Clad inner radius in the gas plenum region	m
Re	Reynolds number	–
R_{ec}	Thermal resistance between clad and coolant	m^2-K/W
R_{ehf}	Thermal resistance of the outer fourth of the cladding	m^2-K/W
r_{erp}	Cladding outer radius in the gas plenum region	m
R_g	Thermal resistance of the gas in the plenum	m^2-K/W
$r_o(i)$	Steady-state radial mesh	m
S_{er}	Perimeter of the cladding or reflector in contact with the coolant	m
S_{st}	Structure perimeter, heat-transfer area per unit height	m
T	Temperature	K
t	Time	s
T_{cin}	Coolant inlet temperature	K
T_{cout}	Coolant outlet temperature	K
T_{eex}	Extrapolated clad temperature	K
T_{eq}	Equilibrium temperature in the mixing volume	K
$\overline{T}_{exp}$	Temperature of the sodium expelled from the subassembly into a mixing volume, averaged over a time step	K
$\overline{T}_f$	Average fuel temperature at an axial node, mass-weighted average	K

continues on next page

Table 3.18.1 – continued from previous page

Symbol	Description	Units
T_g	Plenum gas temperature	K
T_{liq}	Liquidus temperature	K
T_{out}	Bulk temperature in the coolant outlet plenum	K
t_{p1}, t_{p2}	Times at the beginning and end of a PRIMAR time step	s
T_{ri}	Reflector inner node temperature	K
T_{ro}	Reflector outer node temperature	K
T_{sol}	Solidus temperature	K
T_o	Temperature at the beginning of a time step	K
T_1	Temperature at the beginning of a time step	K
T'_1	Temperature of the coolant entering an axial node at the end of a time step	K
T_2	Temperature at the end of a time step	K
T'_2	Temperature of the coolant entering an axial node at the end of a time step	K
U_{melt}	Heat of fusion	J/kg
v	Velocity	m/s
w	Coolant mass flow rate	kg/s
w_e	Estimated mass flow rate	kg/s
w_{fe}	Thickness of the liquid-sodium film left on the cladding after voiding occurs	m
w_{fr}	Thickness of the liquid-sodium film left on the reflector after voiding occurs	m
w_{fst}	Thickness of the liquid-sodium film left on the structure after voiding occurs	m
w_1, w_2	w at beginning and end of a time step	kg/s
Δw	Change in w during a time step	kg/s
x	Distance	m
x_{I1} (JC)	Nodal contribution to I_1	m^{-1}
x_{I2} (JC)	Nodal contribution to I_2	m/kg
x_{I3} (JC)	Nodal contribution to I_3	
x_{I5} (JC)	Nodal contribution to I_5	kg/m^2
z	Axial position	m
z	Elevation	m
Δz	Node height	m
z_{pb}	Elevation at the bottom of the gas plenum	m
z_{pt}	Elevation at the top of the gas plenum	m
z_{pll}	Reference elevation of the coolant inlet plenum	m
z_{plu}	Reference elevation of the coolant outlet plenum	m
α	Heat capacity terms in the matrix equations for radial temperature profiles	J/m-K
α_e	Cladding thermal expansion coefficient	K^{-1}
α_f	Fuel thermal expansion coefficient	K^{-1}
β	Thermal conductivity terms in the matrix equations for radial temperature profiles	J/m-K
γ_c	Fraction of the total heat production that goes directly into the coolant	–
γ_e	Fraction of the total heat production that goes directly into the cladding	–
γ_s	Fraction of the total heat production that goes directly into the structure	–

continues on next page

Table 3.18.1 – continued from previous page

Symbol	Description	Units
γ_2	Ratio of the structure perimeter to the cladding perimeter	–
Δr	Radial node size	m
$\Delta r_{i,j}$	Effective radial distance for heat flow from node I to node j	m
Δt	Time-step size	s
Δz	Axial node height	m
ε	Thermal emissivity	–
θ_1, θ_2	Degree of explicitness or implicitness in the solution	–
ρ	Density	kg/m^3
ρ_c	Coolant density	kg/m^3
ρ_{cin}	Coolant density in the inlet plenum	kg/m^3
ρ_{cout}	Coolant density in the outlet plenum	kg/m^3
(ρc_g)	Density times specific heat of the plenum gas	J/m^3-K
(ρc_r)	Density times specific heat for the reflector	J/m^3-K
ρ_e	Cladding density	kg/m^3
σ	Stefan-Bolzmann constant	W/m^2-K^4
τ	Time constant for flow rate changes	s
τ_c	Condensation heat-transfer time constant	s
τ_{ro}	Time constant for temperature changes in the outer reflector node	s
τ_{sti}	Time constant for temperature changes in the inner structure node	s
μ	Coolant viscosity	Pa-s
$\overline{\mu}$ (JC)	Average value of μ for node JC	Pa-s
Ψ	Source terms in the matrix equations for radial temperature profiles	–

3.19 Appendices

3.19.1 APPENDIX 3.1: Degree of Implicitness for Flow and Temperature Calculations

A typical calculation in SASSYS-1 involves finding a temperature or flow rate at the end of a time step given the values of relevant parameters at the beginning of the step, as well as the values of driving functions at the beginning and end of the step. Linearized finite difference approximations to the relevant differential equations are usually used, and the accuracy of the finite differencing in time will depend both on the size of the time step and on the degree of implicitness in the solution. By picking appropriate values for the degree of implicitness, it is possible to improve the accuracy of the solution.

The differential equations used for temperatures and flow rates are of the form

$$\frac{\mathrm{dy}}{\mathrm{dt}} = f(y,t) \tag{3.19-1}$$

Where y is the flow rate or temperature being calculated, and f is a function of y and time. The finite difference approximation used for Eq. (3.19-1) is

$$\frac{y_2 - y_1}{\Delta t} = \theta_1 f(y_1,t_1) + \theta_2 f(y_2,t_2) \tag{3.19-2}$$

where

t_1 = time at the beginning of the time step

t_2 = time at the end of the time step

y_1 = y at t_1

y_2 = y at t_2

$$\Delta t = t_2 - t_1 \tag{3.19-3}$$

$$\theta_1 + \theta_2 = 1.0 \tag{3.19-4}$$

The parameters θ_1 and θ_2 determines the degree of implicitness of the solution: for a fully explicit solution θ_1 = 1.0 and θ_2 = 0.0, whereas for a fully implicit solution θ_1 = 0.0 and θ_2 = 1.0.

After linearizing, Eq. (3.19-1) can be put in the form

$$\frac{\mathrm{dy}}{\mathrm{dt}} = f(y_1, t_1) + (t - t_1)\frac{\mathrm{df}}{\mathrm{dt}} + (y - y_1)\frac{\mathrm{df}}{\mathrm{dy}} \tag{3.19-5}$$

or

$$\frac{\mathrm{dy}}{\mathrm{dt}} = A + B(t - t_1) + C(y - y_1) \tag{3.19-6}$$

where

$$A = f(y_1, t_1) \tag{3.19-7}$$

$$B = \frac{\mathrm{df}}{\mathrm{dt}} \tag{3.19-8}$$

$$C = \frac{\mathrm{df}}{\mathrm{dy}} \tag{3.19-9}$$

For instance Eq. (3.9-8) for the coolant flow rate in a channel is

$$I_1\frac{\mathrm{dw}}{\mathrm{dt}} = p_\mathrm{b} - p_\mathrm{t} - w^2 I_2 - A'_\mathrm{fL} w I_{3\mathrm{L}} - A'_\mathrm{fr} w |w|^{1+b_\mathrm{fr}} I_{3\mathrm{T}} - w|w| I_4 - gI_5 \tag{3.19-10}$$

Linearizing the w^2, $w|w|^{1+b_\mathrm{fr}}$, and $w|w|$ terms gives

$$w^2 \simeq w_1^2 + 2w_1(w - w_1) \tag{3.19-11}$$

$$w|w|^{1+b_\mathrm{fr}} \simeq w_1|w_1|^{1+b_\mathrm{fr}} + (2 + b_\mathrm{fr})|w_1|^{1+b_\mathrm{fr}}(w - w_1) \tag{3.19-12}$$

and

$$w|w| \simeq w_1|w_1| + |w_1|(w - w_1) \tag{3.19-13}$$

Also, p_b and p_t are approximated as

$$p_\mathrm{b}(t) = p_\mathrm{b}(t_1) + (t - t_1)\frac{\mathrm{dp_b}}{\mathrm{dt}} \tag{3.19-14}$$

$$p_\mathrm{t}(t) = p_\mathrm{t}(t_1) + (t - t_1)\frac{\mathrm{dp_t}}{\mathrm{dt}} \tag{3.19-15}$$

where

$$\frac{\mathrm{dp_b}}{\mathrm{dt}} = \frac{p_\mathrm{b}(t_2) - p_\mathrm{b}(t_1)}{t_2 - t_1} \tag{3.19-16}$$

and

$$\frac{\mathrm{dp_t}}{\mathrm{dt}} = \frac{p_\mathrm{t}(t_2) - p_\mathrm{t}(t_1)}{t_2 - t_1} \tag{3.19-17}$$

Then Eq. (3.19-10) has the form of Eq. (3.19-6) if

$$A = \frac{\left\{p_{\mathrm{b}}\left(t_1\right) - p_{\mathrm{t}}\left(t_1\right) - w_1^2 I_2 - A'_{\mathrm{fL}} w_1 I_{3\mathrm{L}} - A'_{\mathrm{fr}} w_1 \left|w_1\right|^{1+b_{\mathrm{fr}}} I_{3\mathrm{T}} - w_1 \left|w_1\right| \ I_4 - g I_5\right\}}{I_1} \tag{3.19-18}$$

$$B = \frac{\frac{\mathrm{dp_b}}{\mathrm{dt}} - \frac{\mathrm{dp_t}}{\mathrm{dt}}}{I_1} \tag{3.19-19}$$

and

$$C = -\frac{\left[2 w_1 I_2 - A'_{\mathrm{fL}} I_{3\mathrm{L}} - \left(2 + b_{\mathrm{fr}}\right) A'_{\mathrm{fr}} \left|w_1\right|^{1+b_{\mathrm{fr}}} I_{3\mathrm{T}} - \left|w_1\right| I_4\right]}{I_1} \tag{3.19-20}$$

The exact solution for Eq. (3.19-6) is

$$y - y_1 = \left(A\tau - B\tau^2\right)\left(1 - e^{-(t-t_1)/\tau}\right) + B\tau\left(t - t_1\right) \tag{3.19-21}$$

where

$$\tau = -\frac{1}{C} \tag{3.19-22}$$

The finite difference approximation used for Eq. (3.19-6) is

$$\frac{y_2 - y_1}{\Delta t} = A + \theta_2 B \Delta t + \theta_2 C\left(y_2 - y_1\right) \tag{3.19-23}$$

where

$$\Delta t = t_2 - t_1 \tag{3.19-24}$$

The solution of Eq. (3.19-23) for y_2 is

$$y_2 - y_1 = \frac{A\Delta t + \theta_2 B \Delta t^2}{1 + \theta_2 \frac{\Delta t}{\tau}} \tag{3.19-25}$$

Figure 3.19.1 shows y_2 - y_1, as given by Eq. (3.19-25), as a function of θ_2 for the case where A = 1, B = .5, Δt = 2, and τ = 1. Also shown is the exact solution from Eq. (3.19-21). Note that a value θ_2 can be found such that the finite difference solution of Eq. (3.19-25) exactly matches the differential equation solution, given by Eq. (3.19-21), for this case. Also note that neither a fully explicit solution, θ_2 = 0, nor a fully implicit solution, θ_2 = 1.0, is very accurate for particular case. An explicit solution would be numerically unstable for Δt greater than τ; and values considerably smaller than the time constant τ.

For any values of the parameters in Eq. (3.19-21) and Eq. (3.19-25), a value of θ_2 can be chosen such that the finite difference solution matches the solution of the differential equation by setting $t = t_2$ in Eq. (3.19-21) and combining this equation with Eq. (3.19-25) to give

$$\left(A\tau - B\tau^2\right)\left(1 - e^{-\Delta t/\tau}\right) + B\tau\Delta t = \frac{A\Delta t + \theta_2 B \Delta t^2}{1 + \theta_2 \frac{\Delta t}{\tau}} \tag{3.19-26}$$

solving for θ_2 gives

$$\theta_2 = \frac{\frac{\Delta t}{\tau} - \left(1 - e^{-\Delta t/\tau}\right)}{\frac{\Delta t}{\tau}\left(1 - e^{-\Delta t/\tau}\right)} \tag{3.19-27}$$

Note that A and B cancelled out of Eq. (3.19-27), and θ_2 is a function only of $\Delta t/t$.

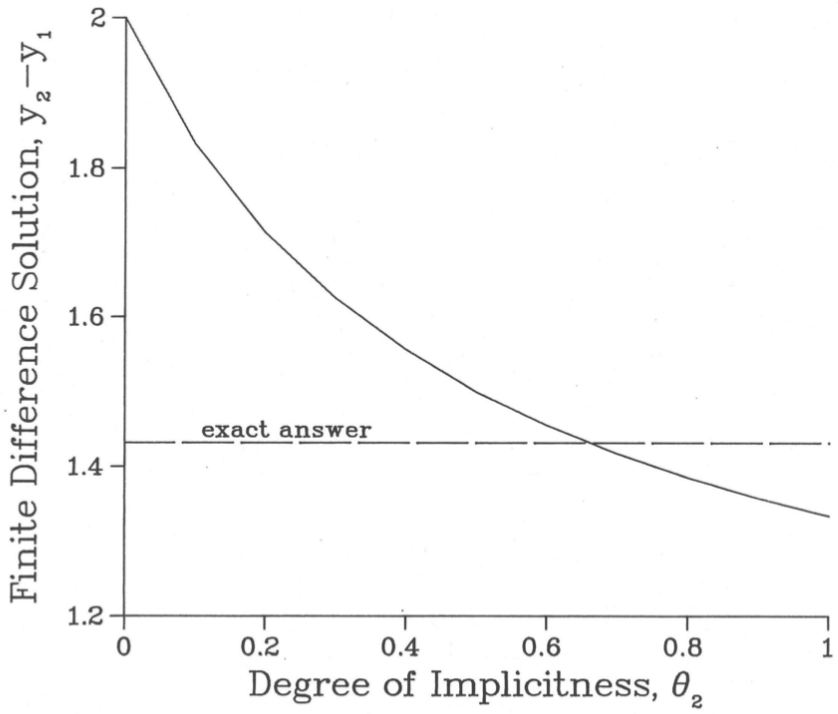


Fig. A3.1-1. Finite Difference Solution as a Function of the Degree of Implicitness

Figure 3.19.1: Finite Difference Solution as a Function of the Degree of Implicitness

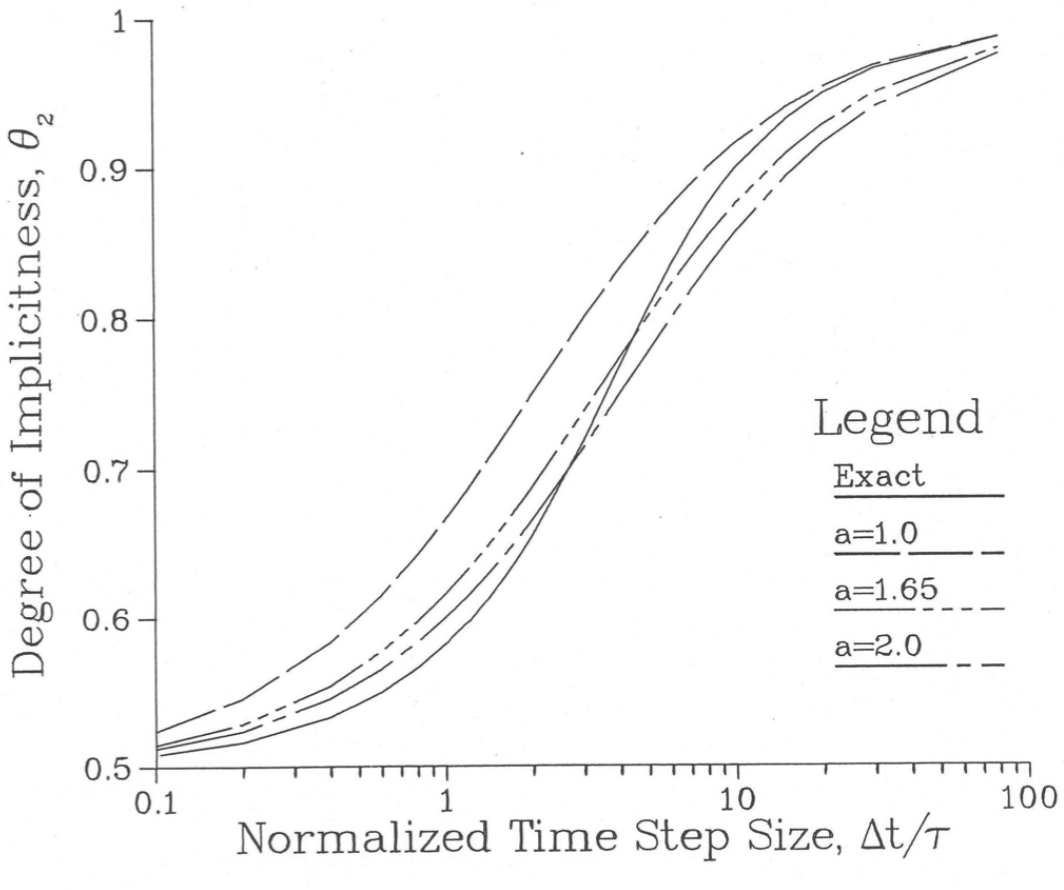


Figure 3.19.2: Degree of Implicitness as a Function of Normalized Time Step Size

Figure 3.19.2 shows θ_2, given by Eq. (3.19-27), as a function of $\Delta t/\tau$. This figure also shows some approximations to this function, as discussed below. For small values of $\Delta t/\tau$, θ_2 approaches 0.5, where as for large values of $\Delta t/\tau$, θ_2 approaches 1.0.

For any equation that can be put in the form of Eq. (3.19-5) or Eq. (3.19-6), Eq. (3.19-22) can be used to find τ, and then Eq. (3.19-27) can be used to find the appropriate value of θ_2 for any value of Δt. Note that if driving pressures and coolant flows are being solved for simultaneously, as is done in PRIMAR-4, then the value of B in Eq. (3.19-6) is not known until the pressures have been solved for. In such cases, a direct analytic solution of the differential equation, as in Eq. (3.19-22), could not be made except by iterating between Eq. (3.19-22) and the pressure solution. On the other hand, the calculation of θ_2, as in Eq. (3.19-27), requires only Δt and τ; and the calculation of τ requires only information that is available at the beginning of a time step; so an appropriate value for θ_2 can be found for use in finite difference approximations without iteration, even if driving pressures and flow rates are being solved for simultaneously.

Even if finite differencing in time does not introduce any error into a solution, there are usually other sources of error, such as the linearization approximations of Eq. (3.19-11) through Eq. (3.19-17). Also, the term represented by $B\,(t - t_1)$ in Eq. (3.19-6) may in fact not be linear in time. Therefore, simpler and easier to compute approximations to the expression on the right-hand side of Eq. (3.19-27) might sometimes be used without losing much overall accuracy. A simple approximation, ${\theta'}_2$ that approaches the correct limits for very small and very large values of $\Delta t/\tau$ is

$$ {\theta'}_2 = \frac{a + \frac{\Delta t}{\tau}}{2a + \frac{\Delta t}{\tau}} \tag{3.19-28} $$

where the parameter a can be chosen to give some best overall fit. Figure 3.19.2 shows this function for various values of the parameter a. The curve for a = 1.65 gives a fairly good fit, although no value of a will give a good fit over the whole range.

A better fit to θ_2 can be obtained by an expression of the form

$$ {\theta'}_2 = \frac{a + bx + x^2}{2a + cx + x^2} \tag{3.19-29} $$

where

$$ x = \frac{\Delta t}{\tau} \tag{3.19-30} $$

with a = 6.12992, b = 2.66054, and c = 3.56284, this expression fits the exact value of Eq. (3.19-27) to within 1% over the whole range of $\Delta t/\tau$, as shown in Figure 3.19.3.

In general, when linear approximations are valid, and when a single known time constant dominates the behavior, Eq. (3.19-27) or Eq. (3.19-29) will give a value of θ_2 that will provide accurate finite difference solutions. For cases where a number of different time constants are important, none of these expressions for θ_2 will give a precise finite difference solution for all time-step sizes. Even in this case, though, θ_2 should approach 0.5 for small time steps, and it should approach 1.0 for large time steps. Therefore, an expression of the form given by Eq. (3.19-28) may be appropriate in such cases.

3.19.2 APPENDIX 3.2: Input Preprocessor

3.19.2.1 Introduction

A preliminary version of an input preprocessor has been written for the three-dimensional core thermal hydraulics model described in Section 3.14. This preprocessor reads a relatively small amount of information from the user and produces a large fraction of the input for the new model. This appendix is split into two main sections. The first section details the calculations carried out by the preprocessor, while the second section is a brief user guide, which details the necessary input and its structure.

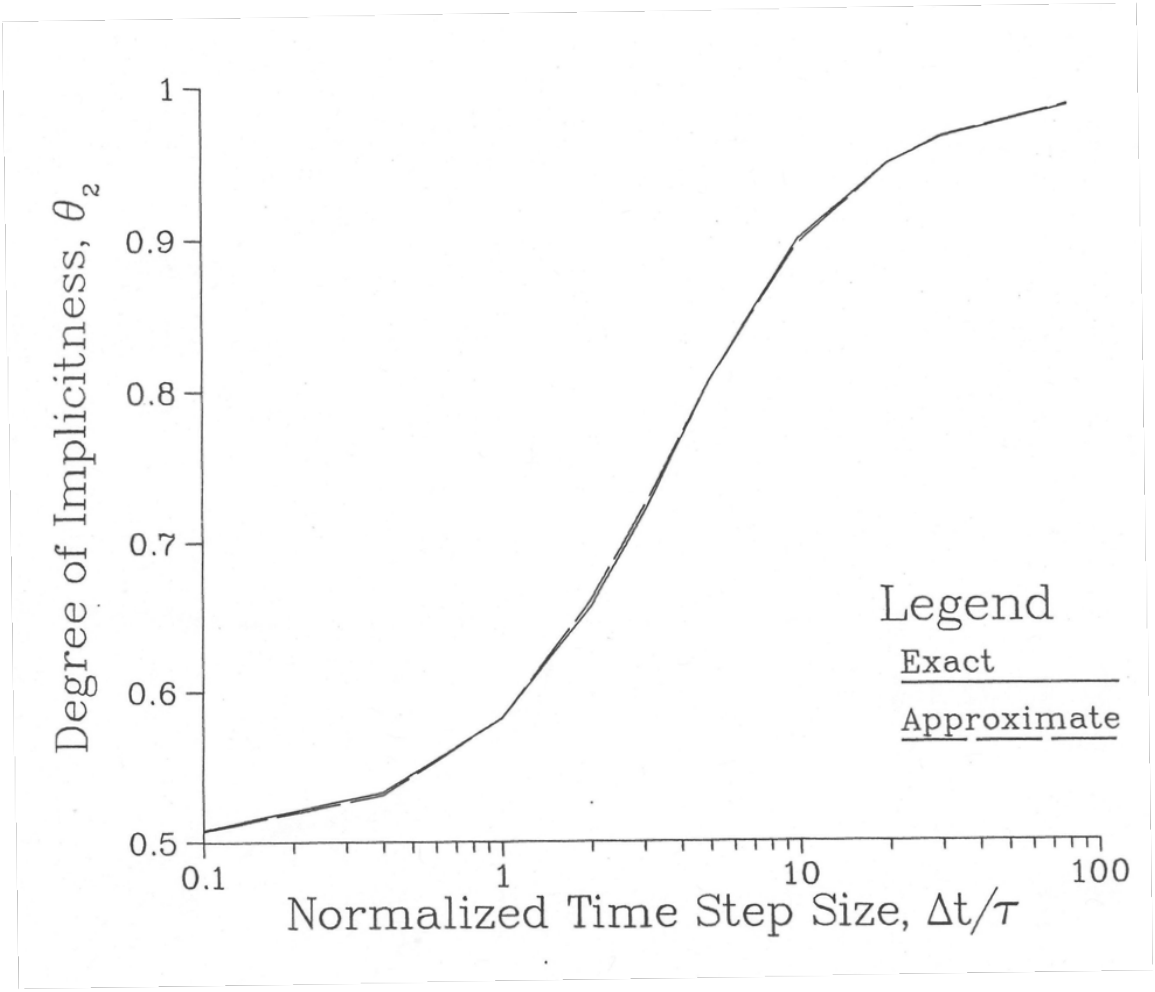


Figure 3.19.3: Approximate Correlation for the Degree of Implicitness

3.19.2.2 Calculation Details

The preprocessor calculates and formats many of the variables necessary for the detailed subchannel model. Some of these variables are calculated by the preprocessor, while others are input directly into the preprocessor. The preprocessor functions can be broken down into three main components, as seen in Figure 3.19.4. First, it calculates geometrical parameters such as coolant flow areas and hydraulic diameters. Second, it uses correlations to compute the parameters used for subchannel-to-subchannel heat flow. Lastly, the parameters used in subassembly-to-subassembly heat transfer are determined. This section details each of these calculations

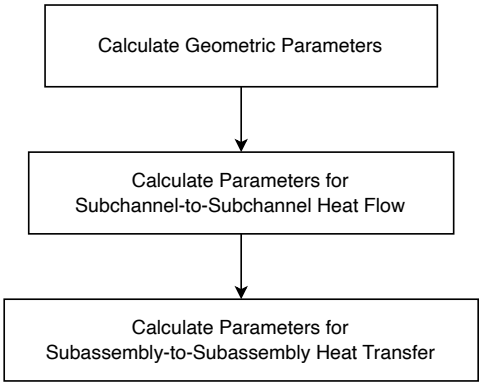


Figure 3.19.4: Preprocessor Coding Overview

3.19.2.2.1 Geometric Properties

This first step to calculating the necessary geometric input for SAS4A/SASSYS-1 is to determine the number of rows[9] of pins in each subassembly. This value is determined from the user-specified number of pins per subassembly n_{SA} (preprocessor variable NPINSB), as shown in Table 3.19.1. Note that if the user input for the number of pins per subassembly n_{SA} (NPINSB) is not listed in Table 3.19.1, the preprocessor will return an error.

Table 3.19.1: Number of Rows of Pins

Number of Pins per Subassembly n_{SA}	Number of Rows of Pins n_{rows}
1	1
7	2
19	3
37	4
61	5
91	6
127	7
169	8
217	9
271	10

[9] A row of pins is the same as a ring of pins

After n_{rows} is calculated, the remaining geometric properties depend on the type of pin treatment specified by the user. As described in Section 3.15.3.1, the subchannel model has several pin options, which are designated using the SAS4A/SASSYS-1 input **JJMLTP**. These different options are selected using the preprocessor input ISC, as shown in Table 3.19.2.

Table 3.19.2: Preprocessor Pin Options ISC

ISC Option	Description
0	JJMLTP = 0, No multi-pin treatment
1	JJMLTP = 1, 1 pin treatment
2	Multi-pin, subchannels grouped by row and sector
3	Multi-pin, same as 2, but bypass flow channel on outside
4	Multi-pin, individual subchannels
5	Multi-pin, individual subchannels, bypass flow channel on the outside
6	JJMLTP = 1, 1 pin treatment, bypass flow channel

The choice of ISC also affects the number of channels used to represent different areas and regions of the subassembly. Table 3.19.3 documents these results, where the values in columns A-E are the number of channels used to represent that region, where Figure 3.19.5 shows the layout of the pin regions. As shown in Table 3.19.3, many of the values for the number of channels in each region depend on n_{rows}, or $n_{c_{\text{ISC}}}$, which is the number of channels for the inner subchannels and can take on value 0 or 6, depending on the input for IPDOPT.

Table 3.19.3: Number of Channels in the Preprocessor Pin Channel Calculation[10]

ISC	A Edge SC	B Corner SC	C Thimble Flow Region	D Represent SA Type	E Inner SC
0				1	1
1				1	1
2	6			$6n_{\text{rows}}$	D-6-$n_{c_{\text{ISC}}}$
3	6		6	$6(n_{\text{rows}}+1)$	D-12-$n_{c_{\text{ISC}}}$
4	$6(n_{\text{rows}}-1)$	6		$2n_{\text{rows}}+4$	D-A-$n_{c_{\text{ISC}}}$1-B
5	$6(n_{\text{rows}}-1)$	6	A-B	$4+8n_{\text{rows}}$	D-A-$n_{c_{\text{ISC}}}$-1-C-B
6			1	2	1

There are several geometric calculation options available using the input IPDOPT, as seen in Table 3.19.4. These options determine how the distance from the duct wall to the center of the first row d_{DC}, the pitch p, and the flat-to-flat inner diameter D_{ff} are calculated. Figure 3.19.6 presents a diagram of these variables.

[10] Equations reference other variable columns using the assigned letter. SC=Subchannel, SA=Subassembly.
[11] Thimble flow region not shown

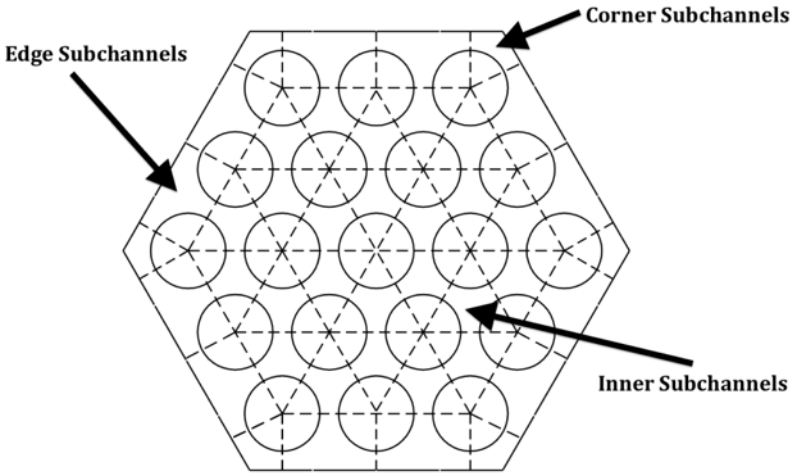


Figure 3.19.5: Pin Region Layout[11]

Table 3.19.4: IPDOPT Options

IPDOPT	Input	Calculated	Description
0	p, D_{ff}	d_{DC}	Use input p everywhere, calculate d_{DC} based on remainder
1	D_{ff}	p, d_{DC}	Use D_{ff}, calculate uniform p, d_{DC}
2	p	d_{DC}, D_{ff}	Use input p, calculate d_{DC} with same slop
3	D_{ff}, p	d_{DC}	Hot channel option: use: $p = D_{\text{ww}} + D_{\text{pin}}$[8] for 1 inner row of subchannels, use p for other inner subchannels, calculate d_{DC} based on remainder
4	p, d_{DC}	D_{ff}	Use input p and d_{DC}, calculate D_{ff}

3.19.2.2.1.1 Basic Dimensions Calculation

Depending on the choice of preprocessor inputs IPDOPT and ISC, the SAS4A/SASSYS-1 input for the channel geometry will be calculated differently. In any case, the basic geometric inputs listed in Table 3.19.5 are then used to calculate additional geometric information, such as those variables listed in Table 3.19.6. Some of these variables are needed for direct SAS4A/SASSYS-1 input, such as **ACCZ**, while others are used to calculate other variables in the following sections, such as the flow splits or subassembly-to-subassembly heat transfer coefficients.

[8] D_{ww} = Wire-wrap diameter, D_{pin} = Pin diameter

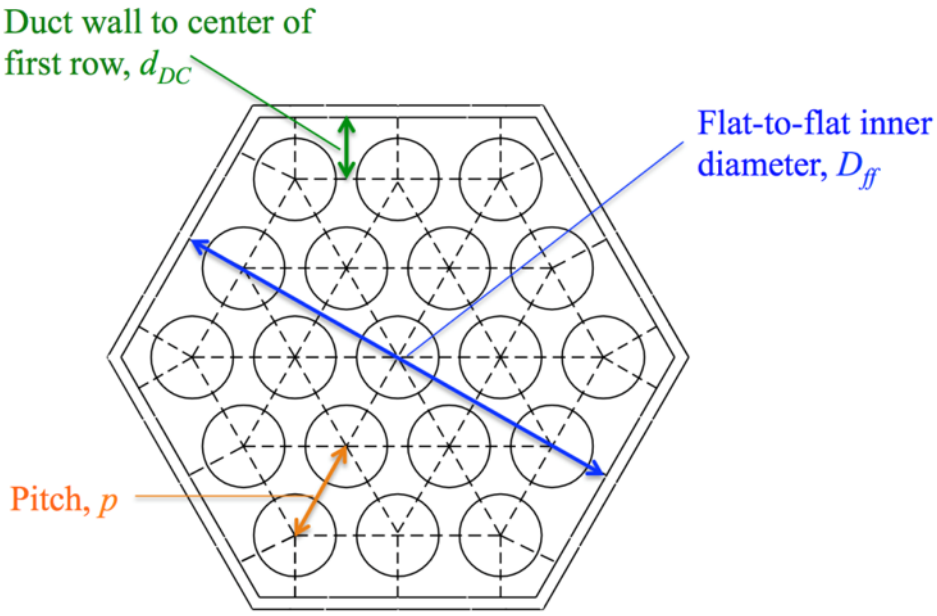


Figure 3.19.6: Assembly Layout Diagram

Table 3.19.5: Preprocessor Input Geometric Variables

Symbol	Input	Description
D_{ff}	FFID	Flat-to-flat inner diameter
d_{DC}	DEDGE	Distance from duct wall to center of 1st row of pins
D_{pin}	DPIN	Diameter of pin
D_{WW}	DWW	Diameter of wrapper wire
p	PITCH	Pin centerline to centerline distance
l_{WW}	HWW	Wrapper wire lead length
t_C	DRE	Thickness of clad
t_{DW}	TDW	Thickness of duct wall
t_{GAP}	TGAP	Thickness of gap between subassemblies

Table 3.19.6: Calculated Geometric Variables

Symbol	Variable	Description
A_{pin}		Area of pin
A_{WW}		Area of wrapper wire
P_{WW}		Perimeter of wrapper wire
A_{hex}		Area of subassembly hex
P_{hex}		Perimeter of subassembly hex
A_{CC}	`ACCZ`	Cross-sectional flow area per pin
D_h	`DHZ`	Hydraulic diameter

3.19.2.2.1.2 Flow Splits Calculation

The flow split calculation starts with establishing the exponential variable α, which is calculated using the preprocessor input BFR. BFR is also used in SAS4A/SASSYS-1 for the calculation of the turbulent friction factor.

$$\alpha = \frac{1 - \text{BFR}}{2 + \text{BFR}} \tag{3.19-31}$$

From there, the flow split per region f_{XX} can be calculated using the following equation, where XX is the region abbreviation seen in Table 3.19.1.

$$f_{\text{XX}} = A_{\text{CC}_{\text{XX}}} D_{\text{h}_{\text{XX}}}^{\alpha} \tag{3.19-32}$$

Table 3.19.7: Region Abbreviations

Abbreviation - XX	Region
INR	Inner Row of Subchannels
INB	Inner Subchannels (except inner row if IPDOPT = 3)
RN	Corner Subchannels
ED	Edge Subchannels

The total flow area can be found from Table 3.19.8, where each value in the right hand column is added to the total flow area.

Table 3.19.8: Total Flow Area Contributions

Region		Amount Added to Total Flow Area
INR		$3f_{\text{INR}}$
INB	If $n_{\text{rows}} = 1$,	f_{INB}
	If $n_{\text{rows}} \neq 1$,	$(3 - 6n_{\text{rows}} + 3n_{\text{rows}}^2 - 3^*) f_{\text{INB}}$
RN		f_{RN}
ED	If $n_{C_{\text{RN}}} > 0$,	$(3\,(n_{\text{rows}} - 1))\, f_{\text{ED}}$
	If $n_{C_{\text{RN}}} = 0$,	$(1 + 3\,(n_{\text{rows}} - 1))\, f_{\text{ED}}$

* Subtract 3 only if the number of inner subchannels > 0

Then the normalized initial coolant flow split $w_{\text{N}_{\text{XX}}}$ is the value of the flow split for region XX divided by the total flow area, determined by Table 3.19.8, times the total core flow w_{core} , which is defined by the preprocessor input WTOTL.

$$w_{\text{N}_{\text{XX}}} = \left(\frac{f_{\text{XX}}}{\text{Total}} \right) w_{\text{core}} \tag{3.19-33}$$

If bypass subchannels (thimble flow corners) are present, the same process holds, where XX is the region from Table 3.19.9.

$$f_{\text{XX}} = A_{\text{CC}_{\text{XX}}} D_{\text{h}_{\text{XX}}}^{\alpha} \tag{3.19-34}$$

Table 3.19.9: Bypass Region Abbreviations

Abbreviation	Region
BYE	Bypass Edge Subchannels
BYC	Bypass Corner Subchannels

The total bypass flow area can be found from Table 3.19.10

Table 3.19.10: Total Bypass Flow Area Contributions

Region		Amount Added to Total Bypass Flow Area
BY	If ISC = 3	$6 f_{\text{BYE}}$
	If ISC = 5	$6\left(n_{\text{rows}}-1\right) f_{\text{BYE}}$
	If ISC = 6	f_{BYE}
NR	If ISC = 3	0
	If ISC = 5	$6 f_{\text{NR}}$
	If ISC = 6	0

The bypass normalized initial coolant flow split is then just the regional flow split divided by the total bypass flow area, multiplied by the total bypass flow w_{bypass}, which is defined by the preprocessor input WBYFL.

$$w_{\text{N}_{\text{XX}}} = \left(\frac{f_{\text{XX}}}{\text{Total}_{\text{bypass}}}\right) w_{\text{bypass}} \tag{3.19-35}$$

3.19.2.2.2 Subchannel-to-Subchannel Heat Flow Coefficients

For the multiple pin options (ISC = 2, 3, 4, or 5), the subchannel-to-subchannel heat flow coefficients must be determined (Eq. (3.14-7) in Section 3.14.2). The SAS4A/SASSYS-1 input **UACHM1**, which is used in the coolant subchannel-to-subchannel heat flow per pin unit height calculation, can be split into coolant and clad contributions, as explained in Section 3.14.2,

$$\text{UACHM1}_{\text{total}} = \text{UACHM1}_{\text{XXXX}} + \text{UACHM1}_{\text{Clad}} \tag{3.19-36}$$

where $\text{UACHM1}_{\text{XXXX}}$ is the coolant contribution from the XXXX region, and $\text{UACHM1}_{\text{Clad}}$ is the contribution from the cladding. The regions are described in Table 3.19.11.

Table 3.19.11: Region Abbreviations

Abbreviation - XXXX	Region
ININ	Inner row to inner row
IAIA	Inner average row to inner average row
IAE	Inner average row to edge
INIA	Inner row to inner average row
EDED	Edge to edge
EDCN	Edge to corner

The clad contribution is independent of the region, and is defined below, where k_{ww} is the thermal conductivity of the wrapper wire (preprocessor input XKSS), M is a multiplier (preprocessor input FU1E), and k_{gap} is the thermal conductivity of the coolant in the gap (preprocessor input XKGAP).

$$\text{UACHM1}_{\text{Clad}} = \frac{12 k_{\text{ww}} t_{\text{c}} M}{\pi k_{\text{gap}} D_{\text{pin}}} \tag{3.19-37}$$

The coolant contribution to **UACHM1** depends on the region, and uses the following equations for each of the regions defined in Table 3.19.11,

$$\text{UACHM1}_{\text{ININ}} = \left(\frac{0.774 D_{\text{ww}}\left(D_{\text{pin}}+D_{\text{ww}}\right)}{\left(\frac{D_{\text{pin}}+D_{\text{ww}}}{\sqrt{3}}\right) D_{\text{pin}}}\right)\left(\frac{D_{\text{ww}}}{D_{\text{pin}}}\right)^{-0.2627} \tag{3.19-38}$$

$$\text{UACHM1}_{\text{IAIA}} = \left(\frac{0.774\,(p - D_{\text{pin}})\,p}{\frac{p}{\sqrt{3}}D_{\text{pin}}}\right)\left(\frac{p - D_{\text{pin}}}{D_{\text{pin}}}\right)^{-0.2627} \tag{3.19-39}$$

$$\text{UACHM1}_{\text{IAE}} = \text{UACHM1}_{\text{IAIA}} \tag{3.19-40}$$

$$\text{UACHM1}_{\text{INIA}} = \left(\frac{0.774 D_{\text{ww}}\,(D_{\text{pin}} + D_{\text{ww}})}{\left(\frac{\frac{p}{\sqrt{3}} + \frac{D_{\text{pin}} + D_{\text{ww}}}{\sqrt{3}}}{2}\right) D_{\text{pin}}}\right)\left(\frac{D_{\text{ww}}}{D_{\text{pin}}}\right)^{-0.2627} \tag{3.19-41}$$

$$\text{UACHM1}_{\text{EDED}} = \frac{1}{2}\left(\frac{0.774\,(2\,(p - D_{\text{pin}}))\,p}{pD_{\text{pin}}}\right)\left(\frac{2\,(p - D_{\text{pin}})}{D_{\text{pin}}}\right)^{-0.2627} \tag{3.19-42}$$

$$\text{UACHM1}_{\text{EDCN}} = \frac{1}{2}\left(\frac{0.774\,(2\,(p - D_{\text{pin}}))\,p}{\left(\frac{p}{2} + \frac{d_{\text{DC}}}{\sqrt{3}}\right) D_{\text{pin}}}\right)\left(\frac{2\,(p - D_{\text{pin}})}{D_{\text{pin}}}\right)^{-0.2627} \tag{3.19-43}$$

The SAS4A/SASSYS-1 input **UACHM2**, which is used in the coolant subchannel-to-subchannel heat flow per pin unit height calculation, can be found by the following (if IPODPT = 3) for inner row to inner row heat flow,

$$\text{UACHM2}_{\text{ININ}} = \frac{\text{CXTC} * D_{\text{ww}}}{2A_{\text{CC}_{\text{INR}}}} \tag{3.19-44}$$

and for the inner row to the inner average, ($\text{UACHM2}_{\text{ININ}}$ and $\text{UACHM2}_{\text{INIA}}$ are zero for IPODPT $\neq 3$)

$$\text{UACHM2}_{\text{INIA}} = \frac{\text{CXTC} * D_{\text{ww}}}{A_{\text{CC}_{\text{INR}}} + A_{\text{CC}_{\text{INB}}}} \tag{3.19-45}$$

For all IPODPT options, the inner average to inner average value follows,

$$\text{UACHM2}_{\text{IAIA}} = \frac{\text{CX} * (p - D_{\text{pin}})}{2A_{\text{CC}_{\text{INB}}}} \tag{3.19-46}$$

For the inner average to the edge,

$$\text{UACHM2}_{\text{IAE}} = \frac{\text{CX} * (p - D_{\text{pin}})}{A_{\text{CC}_{\text{INB}}} + A_{\text{CC}_{\text{ED}}}} \tag{3.19-47}$$

The value for UACHM2 for both edge to edge, and edge to corner heat flow is zero. The placeholder variables are defined below,

$$\text{CXTC} = \text{CMMTC}\left(\text{TANSQR} * \frac{\text{AR1}}{A1\text{PTC}}\right)^{\frac{1}{2}} \tag{3.19-48}$$

$$\text{CX} = \text{CMT}\left(\text{TANSQR} * \frac{\text{AR1}}{A1\text{PTC}}\right)^{\frac{1}{2}} \tag{3.19-49}$$

$$\begin{aligned} \text{CMMTC} &= 0.1\frac{D_{\text{ww}}}{D_{\text{pin}}}^{-\frac{1}{2}} \quad \text{If } n_{\text{SA}} < 7 \\ \text{CMMTC} &= 0.14\frac{D_{\text{ww}}}{D_{\text{pin}}}^{-\frac{1}{2}} \quad \text{If } n_{\text{SA}} \geq 7 \end{aligned} \tag{3.19-50}$$

$$\begin{aligned} \text{CMT} &= 0.1\left(\frac{p - D_{\text{pin}}}{D_{\text{pin}}}\right)^{-\frac{1}{2}} \quad \text{If } n_{\text{SA}} < 7 \\ \text{CMT} &= 0.14\left(\frac{p - D_{\text{pin}}}{D_{\text{pin}}}\right)^{-\frac{1}{2}} \quad \text{If } n_{\text{SA}} \geq 7 \end{aligned} \tag{3.19-51}$$

$$\text{TANSQR} = \frac{1}{\left(\frac{h_{\text{ww}}}{\sqrt{h_{\text{ww}}^2+(\pi(D_{\text{pin}}+D_{\text{ww}}))^2}}\right)^2} \tag{3.19-52}$$

$$\text{AR1} = \frac{\pi\left(D_{\text{pin}} + D_{\text{ww}}\right) D_{\text{ww}}}{6} \tag{3.19-53}$$

$$\text{A1PTC} = \frac{1}{4}p^2\sqrt{3} - \frac{\pi D_{\text{pin}}^2}{8} \tag{3.19-54}$$

For swirl flow, the SAS4A/SASSYS-1 input **XKSWRL**, which is used to calculate swirl flow between subchannels, can be found using the following for the edge to edge calculation,

$$\text{XKSWR}L_{\text{EDED}} = \frac{C1L(p - D_{\text{pin}})}{A_{\text{CC}_{\text{ED}}}\beta} \tag{3.19-55}$$

where,

$$\begin{aligned} \beta &= \frac{1}{2} \quad \text{If ISC} > 3 \\ \beta &= \frac{1}{6} + \frac{1}{2}\left(n_{\text{rows}} - 1\right) \quad \text{If ISC} \leq 3 \end{aligned} \tag{3.19-56}$$

For XKSWRL from the edge to corner,

$$\text{XKSWRL}_{\text{EDCN}} = \text{XKSWRL}_{\text{EDED}} \tag{3.19-57}$$

Lastly, the value for XKSWRL from the corner to edge depends on the selection for ISC,

$$\begin{aligned} \text{XKSQRL}_{\text{CNED}} &= \frac{6(\text{C1L})(p - D_{\text{pin}})}{A_{\text{CC}_{\text{CN}}}} \quad \text{If ISC} > 3 \\ \beta = \text{XKSWRL}_{\text{CNED}} &= \text{XKSWRL}_{\text{EDED}} \quad \text{If ISC} \leq 3 \end{aligned} \tag{3.19-58}$$

Where the placeholder variables are defined below,

$$\text{C1L} = \text{CST}\left(\text{TANSQR}\left(\frac{\left(\frac{3}{2}\right)\pi\left(D_{\text{pin}} + D_{\text{ww}}\right)\left(\frac{D_{\text{ww}}}{6}\right)}{pd_{\text{DC}} - \frac{\pi D_{\text{pin}}^2}{8}}\right)\right)^{\frac{1}{2}} \tag{3.19-59}$$

$$\begin{aligned} \text{CST} &= 0.1\left(\frac{h_{\text{ww}}}{D_{\text{pin}}}\right)^{0.3} \quad \text{If } n_{\text{SA}} < 7 \\ \text{CST} &= 0.75\left(\frac{h_{\text{ww}}}{D_{\text{pin}}}\right)^{0.3} \quad \text{If } n_{\text{SA}} \geq 7 \end{aligned} \tag{3.19-60}$$

The calculation of the SAS4A/SASSYS-1 input **ALATRL**, which is the coolant lateral flow area per pin, depends greatly on the pin options chosen by ISC and IPDOPT, with too many possible combinations to review in this document.

3.19.2.2.3 Subassembly-to-Subassembly Heat Transfer

Before the subassembly-to-subassembly heat transfer calculations can be conducted, a series of more detailed geometric variables must be found that includes many SAS4A/SASSYS-1 inputs. These variables, shown in Table 3.19.12, are calculated using the basic geometric information found in Section 3.19.2.2.1.

Table 3.19.12: Detailed Geometric Variables

Block	Variable	Description
GEOMIN	RER0	Nominal cladding outer radius
	RBR0	Nominal cladding inner radius
	DRF0	Thickness of the outer reflector node
	DRFI	Thickness of the inner reflector node
	SER	Reflector perimeter per pin wetted by coolant
	RBRPL	Cladding inner radius in gas plenum
	RERPL	Cladding outer radius in gas plenum
	AREAPC	Coolant plus pin area per pin
	SRFSTZ	Structure perimeter
	DSTIZ	Thickness of inner structure node
	DSTOZ	Thickness of outer structure node
POWINC	PRSHAP	Ratio of power per subassembly
PMATCH	XKSTIZ	Inner structure thermal conductivity
	XKSTOZ	Outer structure thermal conductivity
	RHOCSI	Density*heat capacity for the inner structure
	RHOCSO	Density*heat capacity for the outer structure
	RHOCR	Density*heat capacity for the reflector
	XKRF	Thermal conductivity of reflector/cladding (in gas plenum)
COOLIN	W0	Steady-state coolant flow rate per fuel-pin

3.19.2.2.3.1 Single-Pin Options

For the single pin treatment with no bypass channel (ISC = 0 or 1), the preprocessor will only calculate subassembly-to-subassembly heat transfer if the channel represents one subassembly. If the channel represents more than one subassembly, this input must be calculated manually. The first step for calculating subassembly-to-subassembly heat transfer is to determine which subassemblies are in contact. Using the core symmetry option ISYM and the number of rows NROW, the preprocessor finds the six adjacent assemblies. The preprocessor resolves whether a subassembly is at an adiabatic boundary.

Once the adjacent subassemblies are known, the SAS4A/SASSYS-1 input related to subassembly heat transfer, shown in Table 3.19.13, is calculated for each flat of the subassembly. This information is then printed to the preprocessor output in the correct format for insertion into a SAS4A/SASSYS-1 input deck.

Table 3.19.13: Subassembly Heat Transfer SAS Inputs

Block	Variable	Description
INPCHN	**NCHCH**	Number of other channels in contact
INPCHN	**ICHCH**	Channel number that is in contact
INPCHN	**IOPCHC**	Subassembly to subassembly heat transfer option
PMATCH	**HACHCH**	Heat-transfer coefficient x area per unit height

For the single pin treatment with bypass channel (ISC = 6), the additional geometric parameters relating to the bypass channel are found, and then the adjacent subassemblies are determined. From there, the process follows a similar path as the single pin options with no bypass channel, where the variables in Table 3.19.13 are found (but taking into account the bypass) and the output is printed.

3.19.2.2.3.2 Multiple-Pin Options

The calculation process is more complicated for the multi-pin options (ISC = 2, 3, 4, or 5). For each row of pins, the detailed geometric variables in Table 3.19.11 are calculated. The power gradient is found first using the preprocessor input IDRGRD. This information, along with the preprocessor input PRSHP0, is necessary to calculate the SAS4A/SASSYS-1 input POWINC:PRSHAP. The formulas used to calculate the variables in Table 3.19.11 differ depending on the choice of IPDOPT and ISC (whether grouped by sector and row or individual subchannels). Where the channel is located, whether it is an inner, edge, or corner channel, also affects the calculation. The location is determined using the number of rows NROW and the choices for IPDOPT and ISC. This is also used to calculate the SAS4A/SASSYS-1 input **JCHMPN**, which denotes what channels are in contact.

The heat flow parameters found in Section 3.19.2.2.2 are then adjusted (normalized) for the different pin nodes. The calculation of the subassembly heat transfer variables in Table 3.19.12 is also more intricate than the single pin models, and depends on whether individual subchannels are being used (ISC = 4, 5) or subchannels grouped by row and sector (ISC = 2, 3).

3.19.2.3 User Guide

3.19.2.3.1 Overview

The current preprocessor is useful for a limited subset of the cases that SAS4A/SASSYS-1 can handle, as shown in Table 3.19.14. The current preprocessor was written only for hexagonal subassemblies with solid duct walls and wire wrapped pins. A modified preprocessor would be needed to handle square arrays, pins with grid spacers instead of wire wraps, or ductless subassemblies. Also, the preprocessor does not handle subassembly-to-subassembly heat transfer involving a channel that represents more than one subassembly.

Table 3.19.14: Scenarios Covered by Preprocessor

Covered	Not Covered
• Hexagonal Subassemblies • Solid Duct Walls • Wire Wrapped Pins	• Square Subassemblies • Ductless Subassemblies • Pins with Grid Spacers

The current preprocessor provides input only for the pin section of a subassembly. Input for the reflector zones above and below the pin section must be input into SAS4A/SASSYS-1 manually, as shown in Figure 3.19.7, but such information only needs to be input once for each subassembly type, as explained in Section 3.15.3.1.

The preprocessor numbers the channels in a consistent manner, and determines subchannel connectivity. In the pin zone the preprocessor calculates three types of parameters. As described in Section 3.19.2.2, it first calculates geometrical parameters, such as coolant flow areas and hydraulic diameters. Second, it uses correlations to calculate the parameters used for subchannel-to-subchannel heat flow. Third, it calculates the parameters used in subassembly-to-subassembly heat transfer.

The preprocessor can also print additional user-specified information directly to the created SAS4A/SASSYS-1 input in order to help reduce the time needed to create the SAS4A/SASSYS-1 input. This allows a block of text to be input once into the preprocessor, which then duplicates that text for all channels of that type. This is done through the ICARD option, and is explained in more detail in Section 3.19.2.2.3 and Section 3.19.2.3.5.

The preprocessor could be of some use even if the three dimensional core thermal hydraulics model is not being used. For a single average channel treatment, the preprocessor can calculate the geometrical parameters such

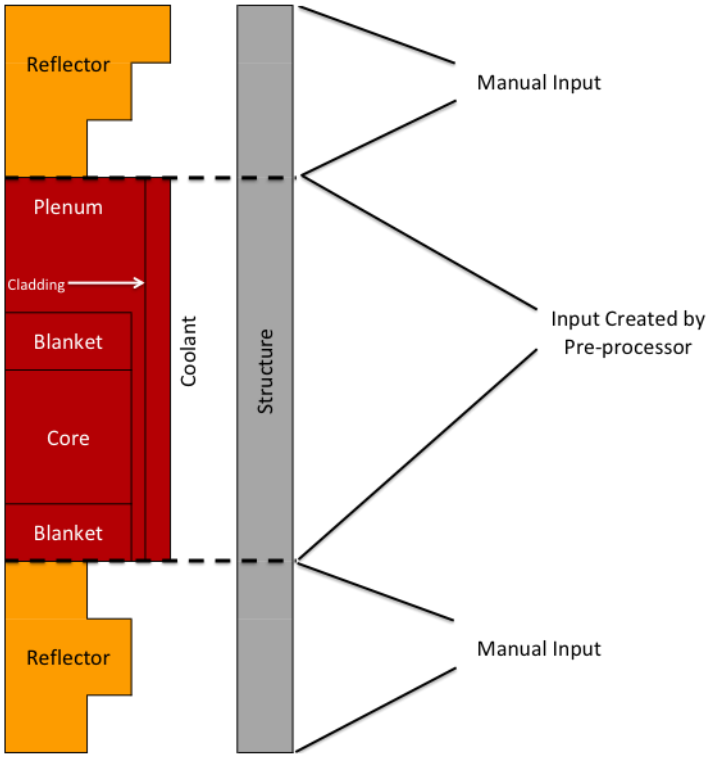


Figure 3.19.7: Preprocessor Coverage

as coolant flow areas and hydraulic diameters for each subassembly. Also, if a separate channel represents each subassembly, then input for subassembly-to-subassembly heat transfer can be calculated by the preprocessor.

3.19.2.3.2 Input/Output Format

The layout of the input file can be broken into three main sections, as shown in Figure 3.19.8. The general input is first, then the subassembly type(s) information, which can also include the optional visualization input, followed by the individual subassembly information. This last section can also include the ICARD information if the preprocessor options ICD or ICDBFR are being used, which allow additional text to be added to the preprocessor output.

Table 3.19.15 provides more detail about the input structure, beginning with the general input. Each subassembly type must have its ITYPE value in consecutive order. Then, ITYPE = 0 is used to signal the end of the subassembly type input. Also, the optional visualization input is entered independently for each subassembly type. A similar structure is used for the subassembly input, where the value for ISUBAS must be in consecutive order, and ISUBAS = 0 signals the end of the subassembly input. The optional ICARD data is entered independently for each subassembly. The format of the input is described in Table 3.19.16.

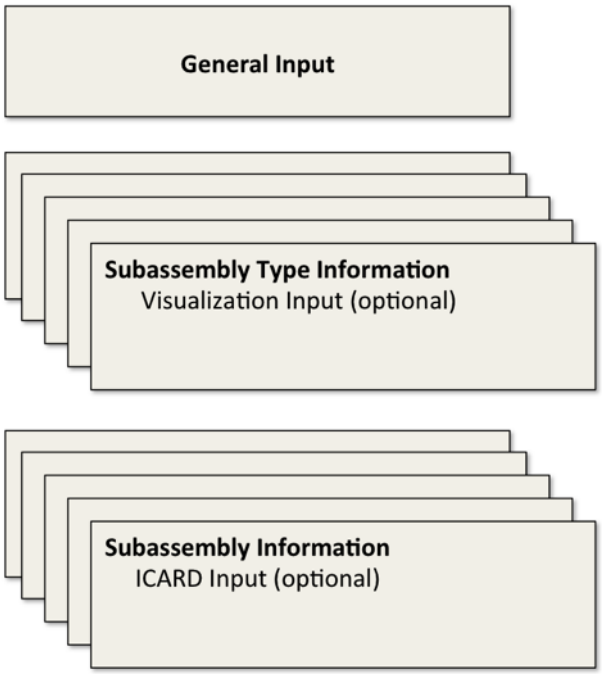


Figure 3.19.8: Preprocessor Input Layout

Table 3.19.15: Preprocessor Input Structure

General Input	
1st Subassembly Type Input (ITYPE = 1)	
	Visualization Input (optional)
2nd Subassembly Type Input (ITYPE = 2)	
	Visualization Input (optional)
…	
Last Subassembly Type Input	
	Visualization Input (optional)
ITYPE 0	
1st Subassembly Input (ISUBAS = 1)	
	ICARD Input (optional)
2nd Subassembly Input (ISUBAS = 2)	
	ICARD Input (optional)
…	
Last Subassembly Input	
	ICARD Input (optional)
ISUBAS 0	

Table 3.19.16: Preprocessor Input Formatting

Section/Variable	Fortran Format
General Input	
ITITLE	(A80)
NROW, ISYM, IFSTCH, IHEXPL, IHEXP2, IDBUG, IFSTSA	7(7X,I5)
Subassembly Type Input	
ITYPTL	(A80)
ITYPE, ISC(ITYPE), NPINSB(ITYPE), IPDOPT(ITYPE), KZPIN(ITYPE), IBYOPT(ITYPE), ISWDIR(ITYPE)	7(7X,I5)
DPIN(ITYPE), DWW(ITYPE), PITCH(ITYPE), DEDGE(ITYPE)	4(10X,F10.5)
FFID(ITYPE), HWW(ITYPE), TBY(ITYPE), TDWBY(ITYPE)	4(10X,F10.5)
AFR(ITYPE), BFR(ITYPE), AFLAM(ITYPE), RELAM(ITYPE)	4(10X,F10.5)
DRE(ITYPE), FU1E(ITYPE), HCBY(ITYPE)	3(10X,F10.5)
TDW(ITYPE), TGAP(ITYPE), RHOCSS(ITYPE), RHOCNA(ITYPE)	4(10X,E10.3)
XKSS(ITYPE), XKGAP(ITYPE), FRSTIN(ITYPE), FRSTED(ITYPE)	4(10X,E10.3)
Visualization Input (optional)	
LDETL(ITYPE), MZ(ITYPE), NT(ITYPE), IEQMAS(ITYPE)	6(7X,I5)
(ZMZ(IZ,ITYPE),IZ=1,MZ)	(6E12.5)
RINFP, ROUTFP	(2E12.5)
Subassembly Input	
ISUBAS, ITYPEV(ISUBAS), KMAX, IDRGRD(ISUBAS), NSUB(ISUBAS), ICD(ISUBAS)	6(7X,I5)
SATITL	(A80)
ICDBFR, IHTLAT(ISUBAS)	2(7X,I5)
WTOTL(ISUBAS), PRSHP0(ISUBAS), PGRAD(ISUBAS), ADOPPN(ISUBAS)	4(10X,F10.6)
BDOPPN(ISUBAS), WBYFL(ISUBAS), HALATP(ISUBAS)	3(10X,F10.6)
(IHXPSS(K), IHXPSE(K),K=1,KMAX)	(20X,12I5)
NPRINT, (ICHPRT(I,ISUBAS),I=1,NPRINT)	(20x,12I5)

3.19.2.3.3 Input Description

As stated in Section 3.19.2.3.2 and shown in Table 3.19.17, the input for the preprocessor can be split into three major sections with the visualization input under the subassembly type section. The following tables outline the input for each section. The input needed for the visualization tool, described in Section 3.19.3, is also provided.

Table 3.19.17: Preprocessor Input Sections

Section	Description
General Input	General Input
Subassembly Type Input	Specifies pin treatment and geometric options for each subassembly type
Visualization Input (optional)	Specifies details for creation of visualization mesh, described in Section 3.19.3
Subassembly Input	Subassembly specific data

Table 3.19.18: General Input

Symbol	Description
ITITLE	One line of title information. (80 character limit)
NROW	Number of rows of subassemblies in the model.
ISYM	Symmetry information = 1, Full core = 2, Half core = 3, Third core = 4, Quarter core = 6, Sixth core = 12, 1/12 core
IFSTCH	First channel number to use.
IHEXPL	Plot information = 0, Skip plot of subassembly hexes numbered with hex positions for the symmetry and number of rows being used = 1, Plot to file HEX_NUM.ps and continue = 2, Plot to file HEX_NUM.ps and quit
IHEXP2	Plot information = 0, Skip plot of subassembly hexes numbered with subassembly numbers > 0, Plot subassembly hexes numbered with subassembly numbers to file SUB_NUM.ps. **= 2, Generate visualization geometry file for use with VisIt,** additional input must be provided as described in Section 3.19.3
IDBUG	Debugging output > 0, Debugging output
IFSTSA	First subassembly number to use.

Table 3.19.19: Subassembly Type Input

Symbol	Description
ITYPTL	One line of type title information. (80 character limt)
ITYPE	Subassembly type. If ITYPE=0, this is the end of the type input.
ISC (ITYPE)	Pin treatment option = 0, JJMLTP = 0, No multi-pin treatment = 1, JJMLTP = 1, 1 pin treatment = 2, Multi-pin, subchannels grouped by row and sector = 3, Multi-pin, same as 2, but bypass flow channel on outside = 4, Multi-pin, individual subchannels = 5, Multi-pin, individual subchannels, bypass flow channel on the outside = 6, JJMLTP = 1, 1 pin treatment, bypass flow channel
NPINSB (ITYPE)	Number of pins per subassembly.

continues on next page

Table 3.19.19 – continued from previous page

Symbol	Description
IPDOPT (ITYPE)	Geometry Calculation Options = 0, Use input PITCH everywhere, calculate DEDGE based on what is left over, use input FFID = 1, Use FFID, calculate uniform PITCH, DEDGE = 2, Use input PITCH, calculate DEDGE with same slop, calculate FFID = 3, Hot channel option: use FFID, use: P = DPIN + DWW for 1 inner row of subchannels, use PITCH for other inner subchannels, calculate DEDGE based on what is left over = 4, Use PITCH and DEDGE, calculate FFID
KZPIN (ITYPE)	Zone KZ
IBYOPT (ITYPE)	Bypass information = 0, Bypass flow the whole length of the subassembly = 1, Bypass flow only in the pin section Note: IBYOPT is only relevant if ISC = 3, 5, or 6. IBYOPT = 1 is not currently implemented
ISWDIR (ITYPE)	Swirl information = 1, Swirl flow is counter-clockwise, when viewed from above = 2, Swirl flow is clockwise
DPIN (ITYPE)	Pin Diameter.
DWW (ITYPE)	Wrapper wire diameter.
PITCH (ITYPE)	Pin Centerline to Centerline Distance.
DEDGE (ITYPE)	Distance from duct wall to center of first row of pins.
FFID (ITYPE)	Flat-to-flat ID.
HWW (ITYPE)	Wire wrap lead length.
TBY (ITYPE)	Thickness of bypass flow gap.
TDWBY (ITYPE)	Thickness of the outer duct wall beyond the bypass gap.
AFR (ITYPE)	Turbulent Friction Factor = AFR*Re**BFR
BFR (ITYPE)	Turbulent Friction Factor = AFR*Re**BFR
AFLAM (ITYPE)	Laminar Friction Factor = AFLAM/Re
RELAM (ITYPE)	Reynolds number for transition from turbulent to laminar. If RELAM = 0, then it is calculated so that turbulent friction factor equals laminar friction factor at RELAM.
DRE (ITYPE)	Clad thickness.

continues on next page

Table 3.19.19 – continued from previous page

Symbol	Description
FU1E (ITYPE)	Geometry multiplier for subchannel-to-subchannel heat transfer by conduction through cladding. Typical value near 2.0.
HCBY (ITYPE)	Coolant heat transfer coefficient from outer surface of duct wall to thimble flow coolant. For laminar flow, HCBY = 7.6*k/Dh to 8.23*k/Dh.
TDW	Thickness of duct wall.
TGAP	Thickness of gap between subassemblies.
RHOCSS	Density × heat capacity of the wrapper wire and duct wall.
RHOCNA	Density × heat capacity of the coolant in the gap.
XKSS	Thermal conductivity of the wrapper wire and duct wall.
XKGAP	Thermal conductivity of the coolant in the gap.
FRSTIN	Fraction of structure thickness in the inner node, inner subchannel.
FRSTED	Fraction of structure thickness in the inner node, edge subchannel.
Visualization Input (if IHEXP2 = 2)[12]	
LDETL	Level of detail in the visualization mesh geometry file. = 0, Do not generate geometry for this assembly type. = -1, Only simplified coolant subchannels are defined. = -2, Simplified coolant subchannels and duct wall structure. = 1, Detailed coolant subchannels are defined. = 2, Detailed coolant subchannels and duct wall structure. = 3, Detailed subchannels, duct wall, and simple fuel pin. = 4, Detailed subchannels, duct wall, and detailed fule pin.
MZ	Number of axial segments in zone KZPIN (e.g., the MZ mesh).
NT	Number of radial temperature nodes within the fuel (only relevant for LDETL = 4).
IEQMAS	Radial fuel mesh size assumptions (relevant for LDETL=4). = 0, Equal radial distance. > 0, Equal cylindrical area.
ZMZ	Axial segment boundaries for zone KZPIN, where the lowest boundary is assumed to be zero. Note that these and the above values are only used to generate the geometry file, and are not used to define NZNODE, AXHI, or ZONEL.
RINFP	Fuel inner radius.
ROUTFP	Fuel outer radius (default = clad inner radius)

[12] These parameters are only used to generate the visualization mesh geometry and, with the exception of LDETL, are not incorporated into the SAS4A/SASSYS-1 input that is generated for the subchannel definitions.

Table 3.19.20: Subassembly Input

Symbol	Description
ISUBAS	Subassembly number. Max = NSBMAX = 1000. If ISUBAS = 0, then this is the end of the subassembly input.
ITYPEV (ISUBAS)	ITYPE = subassembly type.
KMAX	Highest pin axial node, as shown in Figure 3.15.4
IDRGRD (ISUBAS)	Direction of power gradient. = 1, N(high) - S(low) = 2, NNE - SSW = 3, NE - SW = 4, E - W = 5, SE - NW = 6, SSE - NNW
NSUB (ISUBAS)	Number of real subassemblies represented by this ISUBAS
ICD	If > 0, Read in cards, ICARD, to be put at the end of the first channel in ISUBAS
SATITL (ISUBAS)	Subassembly title information.
ICDBFR	If > 0, Read in cards to be put before the first channel in ISUBAS.
IHTLAT	If > 0, Use HALATP(ISUBAS) for lateral heat transfer across pins. Currently, only implemented for the detailed subchannel treatment (ISC = 4 or 5). Not implemented for the one pin per subassembly case.
WTOTL (ISUBAS)	Total subassembly coolant flow, excluding thimble flow.
PRSHP0 (ISUBAS)	Average PRSHAP for the subassembly. PRSHAP = Ratio of power per subassembly averaged over this channel to the power per subassembly averaged over all channels.
PGRAD (ISUBAS)	Power gradient. PRSHAP $= \frac{\text{PRSHP0}(1+\text{PGRAD})\text{XPIN}}{\text{NPINSB}}$ at SA[13] Edge PRSHAP $= \frac{\text{PRSHP0} \times \text{XNPIN}}{\text{NPINSB}}$ at SA Center PRSHAP $= \frac{\text{PRSHP0}(1-\text{PGRAD})\text{XPIN}}{\text{NPINSB}}$ at SA Other Edge
ADOPPN	ADOP per pin. ADOP = Doppler coefficient for this channel when part of the core represented by this channel is not voided.
BDOPPN	BDOP per pin. BDOP = Doppler coefficient for this channel when part of the core represented by this channel is fully voided.
WBYFL	Total thimble flow. Used only if ISC = 3, 5, or 6.
HALATP	Heat transfer coefficient × area/height for lateral heat transfer across pins.
IHXPSS(K)	Subassembly ISUBAS represents subassemblies with hex position numbers from IHXPSS(K) to IHXPSE(K)
IHXPSE(K)	See above
NPRINT	Number of channels in this subassembly which will be printed out in normal prints (set IPRSKP = 0, in all other channels IPRSKP will be set to 1).
ICHPRT	Relative channel number (1 for the first channel in the subassembly).
ICARD	Information to be read and directly printed to fort.1 output. See ICD and ICDBFR.

3.19.2.3.4 Input Example

An example input is provided in Figure 3.19.9, where "#" is used as a placeholder for user specified values. This example has two subassembly types, and two subassemblies. The visualization tool is also used for both subassembly types. The two subassemblies use the ICARD option, with the first subassembly (ISUBAS = 1) using ICDBFR, and the second subassembly (ISUBAS = 2) using ICD.

3.19.2.3.5 Output Description

Lastly, several output files are created by the preprocessor, as seen in Table 3.19.21. The first output file is the direct output from the preprocessor, which documents notifications printed during its execution. The second file, fort.1 is the SAS4A/SASSYS-1 input created by the preprocessor. If the ICARD option is being used (ICD or ICDBFR > 0), then ICARD information will be printed directly to fort.1. The location of the ICARD information will be before the first channel in ISUBAS if ICDBFR is being used, or at the end of the first channel in ISUBAS if ICD is used.

The preprocessor can optionally produce postscript plots to HEX_NUM.ps and SUB_NUM.ps. HEX_NUM.ps plots the subassembly hexes numbered with the position numbers used internally by the preprocessor. SUB_NUM.ps plots the subassembly hexes numbered with the subassembly numbers assigned by the user. The two plots could be identical if the user picks the same numbering scheme that the preprocessor uses internally. Lastly, the optional visualization geometry file, SAS.sasgeom, can be created. More details on this file can be found in Section 3.19.3.

Table 3.19.21: Output File Description

Output	Description
*.out	Log
fort.1	SAS4A/SASSYS-1 input
HEX_NUM.ps	Plot of the subassembly hexes numbered with the position numbers used internally by the preprocessor (postscript)
SUB_NUM.ps	Plot of the subassembly hexes numbered with the subassembly numbers assigned by the user (postscript)
SAS.sasgeom	Visualization mesh file (optional)

3.19.3 APPENDIX 3.3: Visualization

3.19.3.1 Introduction

An advanced visualization capability has been developed for the SAS4A/SASSYS-1 code [1] that allows for the generation, display, and animation of large datasets that are calculated during a transient simulation. The key features of the new capability include its support for multiple-pin, subchannel modeling, integration with the subchannel input preprocessor, geometry visualization, fully-detailed temperature field support with unlimited dataset sizes, and interactive visualization and animation generation on a desktop PC using the visualization tool VisIt [2].

The motivation for developing an advanced visualization capability results from the ever-increasing fidelity with which fast reactor transient simulations are being carried out and the enormous amounts of data generated during those simulations. For example, the EBR-II SHRT-17 test has recently been modeled [3] with subchannel-level detail and a coarse axial mesh for the experimental assembly XX09 and its six nearest neighbors. For a simulation

[13] SA = Subassembly

```
Title Information (ITITLE)
 NROW      # ISYM      1 IFSTCH    # IHEXPL    1 IHEXP2    2 IDBUG     0 IFSTSA    #
 Subassembly Type 1 (ITYPTL)
 ITYPE     1 ISC       4 NPINSB    # IPDOPT    1 KZPIN     2 IBYOPT    0 ISWDIR    #
 DPIN      #.######  DWW       #.######  PITCH               DEDGE
 FFID      #.######  HWW       #.######  TBY                 TDWBY
 AFR       #.######  BFR      -#.######  AFLAM    #.######   RELAM     #.######
 DRE       #.######  FU1E      #.######  HCBY
 TDW       #.###E#   TGAP      #.###E#   RHOCSS    #.###E#   RHOCNA    #.###E#
 XKSS      #.###E#   XKGAP     #.###E#   FRSTIN    #.###E#   FRSTED    #.###E#
 LDETL     4 MZ       20 NT       10 IEQMAS    0
     0.04000     0.08000     0.12000     0.16000     0.20000     0.24000
     0.28000     0.32000     0.36000     0.40000     0.44000     0.48000
     0.52000     0.56000     0.60000     0.64000     0.68000     0.72000
     0.76000     0.80000
     0.00000     0.00000
 Subassembly Type 2 (ITYPTL)
 ITYPE     2 ISC       4 NPINSB  ### IPDOPT    1 KZPIN     2 IBYOPT    0 ISWDIR    #
 DPIN      #.######  DWW       #.######  PITCH               DEDGE
 FFID      #.######  HWW       #.######  TBY                 TDWBY
 AFR       #.######  BFR      -#.######  AFLAM    #.######   RELAM     #.######
 DRE       #.######  FU1E      #.######  HCBY
 TDW       #.###E#   TGAP      #.###E#   RHOCSS    #.###E#   RHOCNA    #.###E#
 XKSS      #.###E#   XKGAP     #.###E#   FRSTIN    #.###E#   FRSTED    #.###E#
 LDETL     4 MZ       20 NT       10 IEQMAS    0
     0.04000     0.08000     0.12000     0.16000     0.20000     0.24000
     0.28000     0.32000     0.36000     0.40000     0.44000     0.48000
     0.52000     0.56000     0.60000     0.64000     0.68000     0.72000
     0.76000     0.80000
     0.00000     0.00000
 End of Subassembly Type Input:
 ITYPE    0
 ISUBAS    1 ITYPEV    1 KMAX      1 IDRGRD    # NSUB      1 ICD       0
 Subassembly Information 1 (SATITL)
 ICDBFR    1 IHTLAT    #
 WTOTL     #.######  PRSHP0  #.######    PGRAD    #.######   ADOPPN   -#.######
 BDOPPN   -#.######  WBYFL               HALATP   #.######
 IHXPSS, IHXPSE         1    1
 NPRNT, ICHPRT(I)       1    1
 Read in cards, ICARD, to be put before the first channel in ISUBAS (Since ICDBFR = 1)
ENDICD
 ISUBAS    2 ITYPEV    2 KMAX      1 IDRGRD    # NSUB      1 ICD       1
 Subassembly Information 2 (SATITL)
 ICDBFR    0 IHTLAT    #
 WTOTL     #.######  PRSHP0  #.######    PGRAD    #.######   ADOPPN   -#.######
 BDOPPN   -#.######  WBYFL               HALATP   #.######
 IHXPSS, IHXPSE         1    1
 NPRNT, ICHPRT(I)       1    1
 Read in cards, ICARD, to be put at the end of the first channel in ISUBAS (Since ICD =
1)
ENDICD
 End of Subassembly Input:
 ISUBAS    0
```

Figure 3.19.9: Preprocessor Example Input

that lasts 250 seconds with data saved every one-tenth of a second, the full temperature field in the active core region for these seven assemblies is estimated to be approximately 15 GB.[14] While it is trivial to identify maxima and minima within this large dataset, identifying other features is much more difficult. Items of interest might include the identification of power, flow, or temperature fluctuations caused by instabilities in the numerical solution, regions of temperature near (but not at) the peak, flow reversals at the subchannel level, unusual results caused by input error, time-derivatives of solution variables, and so on.

The following sections describe changes made to the subchannel input preprocessor to support creation of visualization mesh geometry and preliminary changes made to the SAS4A/SASSYS-1 code to support large dataset generation. Finally, examples of using the visualization tool VisIt to render three-dimensional images from the SHRT-17 simulation are provided.

3.19.3.2 Mesh Generation

Channel modeling in the SAS4A/SASSYS-1 code is fundamentally based on a single-pin representation that includes associated coolant and structure. Geometric input to the SAS4A/SASSYS-1 code is represented by generalized parameters such as flow area, hydraulic diameter, heat-transfer area, etc, and details of the problem-specific geometry are not fully known. In order to project computed transient data into three-dimensional space, a visualization mesh representing the actual geometry of the problem needs to be defined.

Detailed core models using the multiple-pin subchannel capability in the SAS4A/SASSYS-1 safety analysis code are usually created using the subchannel input preprocessor, detailed in Section 3.19.2. The input preprocessor takes subassembly design information and creates detailed SAS4A/SASSYS-1 input descriptions of the multiple-pin subchannels in terms of geometric, thermal, and hydraulic parameters. Because the input preprocessor contains subassembly design information that is not available to the SAS4A/SASSYS-1 code, it was modified to act as the tool to be used in creating the visualization mesh geometry. A visualization mesh is defined so that computed data can be mapped into three-dimensional space. Furthermore, the mesh can be viewed independent of computed data to identify subchannel locations in the global geometry. Details on the format of the visualization mesh geometry file are available at the end of this appendix.

An example of the visualization mesh created for the SHRT-17 simulation is shown in Figure 3.19.10, where the mesh is colored according to the type of geometry present. In this case, gray represents cladding and structure, blue represents sodium coolant, orange is an average fuel region, and red is a detailed fuel region. The bypass flow area of the experimental assembly XX09 is visible in the assembly located in the upper-right portion of the figure.

Because of the potential for the enormous amounts of data that need to be generated in order to fill a visualization mesh, an option is provided to control the level of detail represented in the visualization mesh geometry file on an assembly-by-assembly basis. Figure 3.19.10 displays two of the many detail options available. In Figure 3.19.10, experimental assembly XX09 is represented by a detailed radial pin mesh that includes nine radial temperature zones in the fuel region. The surrounding assemblies, however, are represented by an average fuel region that includes only a single radial zone for the bulk average fuel temperature. This reduces the data file for the 250 second simulation from the 15 GB estimate described earlier down to approximately 3.5 GB. The different options for the level of detail in the visualization mesh geometry file are listed in Table 3.19.22 and examples of the mesh for each option are shown in Figure 3.19.11.

The detail options shown in Figure 3.19.11 include two very simplified subchannel representations that use a single cell volume for each subchannel (LDETL = -1 or -2). For the interior subchannels, triangular cells are defined; and for the edge and corner subchannels, quadrilateral cells are defined. For LDETL = -1, these cells (artificially) extend over the fuel pin and structure regions in order to occupy the full assembly volume. For LDETL = -2, the cells only extend over the fuel pin regions, and additional cells are defined to represent the duct wall structure regions.

[14] Animations of the SHRT-17 simulation can be viewed at http://internal.ne.anl.gov/~fanning/SHRT17.html

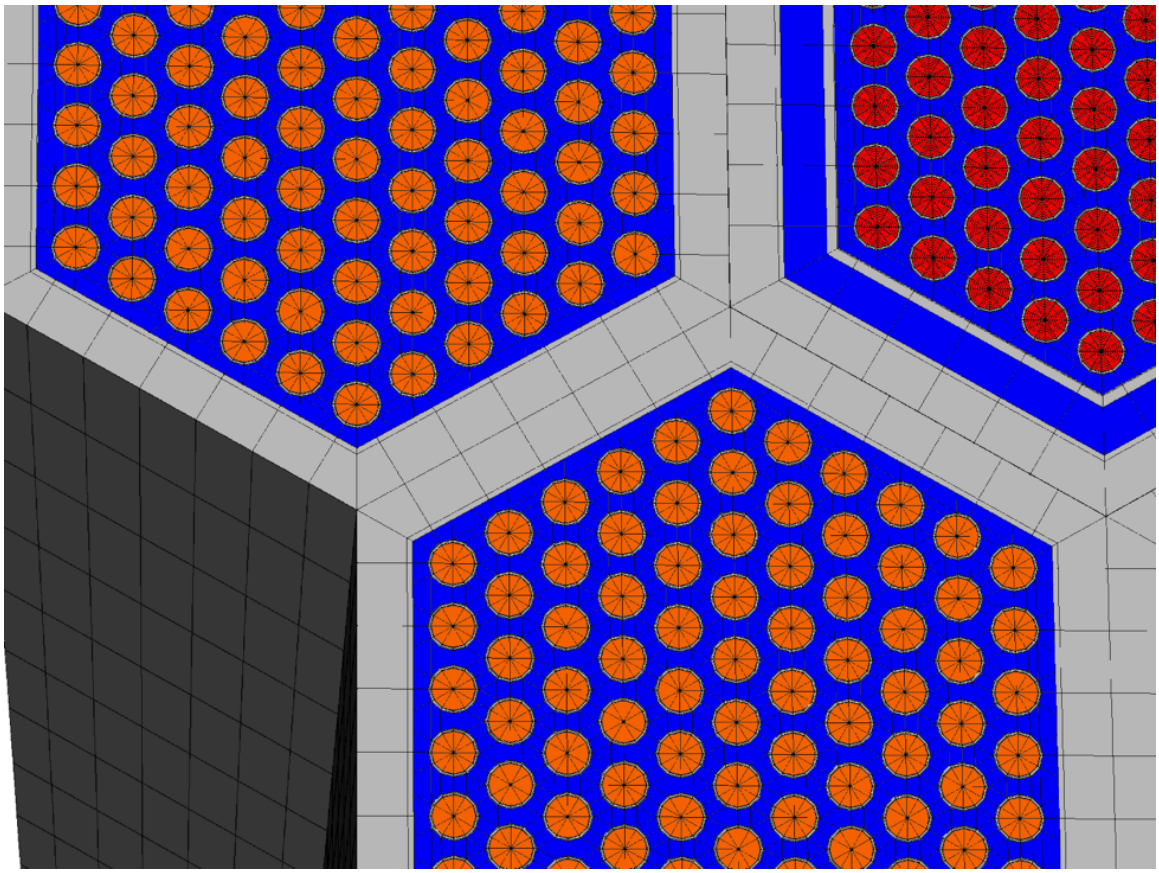


Figure 3.19.10: Pseudo Color Plot of Steady-State Coolant, Structure, and Fuel Temperatures (°C) for XX09 and Surrounding Assemblies.

Positive values of LDETL define multiple coolant cells to approximate the actual shape of the coolant subchannel. LDETL = 1 and LDETL = 2 have void regions in the geometry since neither define regions for fuel pins and the former does not define regions for the duct wall structure. The advantage of LDETL = 1 and LDETL = 2 is that they require the same data as with the corresponding negative values for LDETL but they more accurately represent the physical geometry. LDETL options 3 and 4 introduce regions defined for the fuel pin, including fuel and cladding. The difference between the two options is that LDETL = 3 represents the fuel region by a single radial zone, while LDETL = 4 represents the fuel region by multiple radial zones corresponding to the parameter NT. (See below.)

Table 3.19.22: Geometry Detail Options for Creating Visualization Mesh Geometry Files.

LDETL	Geometric Detail
0	No Geometry Defined
-1	Simplified Coolant Subchannels
-2	Simplified Coolant Subchannels and Duct Wall Structure
1	Detailed Coolant Subchannels
2	Detailed Coolant Subchannels and Duct Wall Structure
3	Detailed Coolant Subchannels, Duct Wall Structure, and Simplified Fuel Pin
4	Detailed Coolant Subchannels, Duct Wall Structure, and Detailed Fuel Pin

It should be noted that even though the coolant subchannels are defined by multiple cells for positive values of LDETL (four in the case of the interior subchannels) SAS4A/SASSYS-1 still only calculates a single temperature for each axial position along a subchannel and that temperature is applied to all cells that belong to a given subchannel. Therefore, it is important to recognize the distinction between the *visualization* mesh described here and the *computation* mesh (i.e. channel model) used by the SAS4A/SASSYS-1 code.

In the multiple-pin subchannel preprocessor, all new parameters associated with the generation of a visualization mesh geometry file are confined to the assembly type definition input. This includes the geometric detail option parameter LDETL. Therefore, even if two assemblies are of the same *type* (both driver assemblies, for example) different types will need to be defined if a different level of geometric detail is desired for each. In the modified preprocessor, the additional parameters are expected only if the existing parameter IHEXP2 = 2. This assures compatibility with existing input files created for earlier versions of the preprocessor. When IHEXP2 = 2, the additional visualization input, shown in Section 3.19.2.3.3, must be added to the end of each assembly type definition.

Using the geometry information contained in the preprocessor input file, along with the additional visualization input shown in Section 3.19.2.3.3, the multiple-pin subchannel preprocessor can create a SAS4A/SASSYS-1 visualization mesh geometry file. By default, the generated file is named "SAS.sasgeom". Although the basename of the file is not critical, the file extension (*.sasgeom) is used by the VisIt visualization software to identify the correct file reader with which to parse the mesh.

The VisIt visualization software can be used to view the visualization mesh geometry file created by the preprocessor without needing to carry out a simulation to generate data. VisIt has the capability to plot numerous qualities of the visualization mesh, including determining individual cell volumes. In addition to the visualization mesh geometry, the file contains two parameters that can be plotted: channel ID and channel type. Channel ID directly corresponds to the SAS4A/SASSYS-1 channel number, and can be used to identify specific channels in the global geometry. Channel type is somewhat of a misnomer,[15] in that it does not define a type for a given channel but rather a region to which a cell in the channel's geometry belongs. (Figure 3.19.10 was created by plotting the channel type parameter.) The currently defined types include

- 0: Coolant (always present)

[15] The misnomer arises because the channel ID and channel type parameters actually exist in the geometry file as a single integer parameter that represents a unique "data channel" that is used to associate simulation data with the visualization mesh.

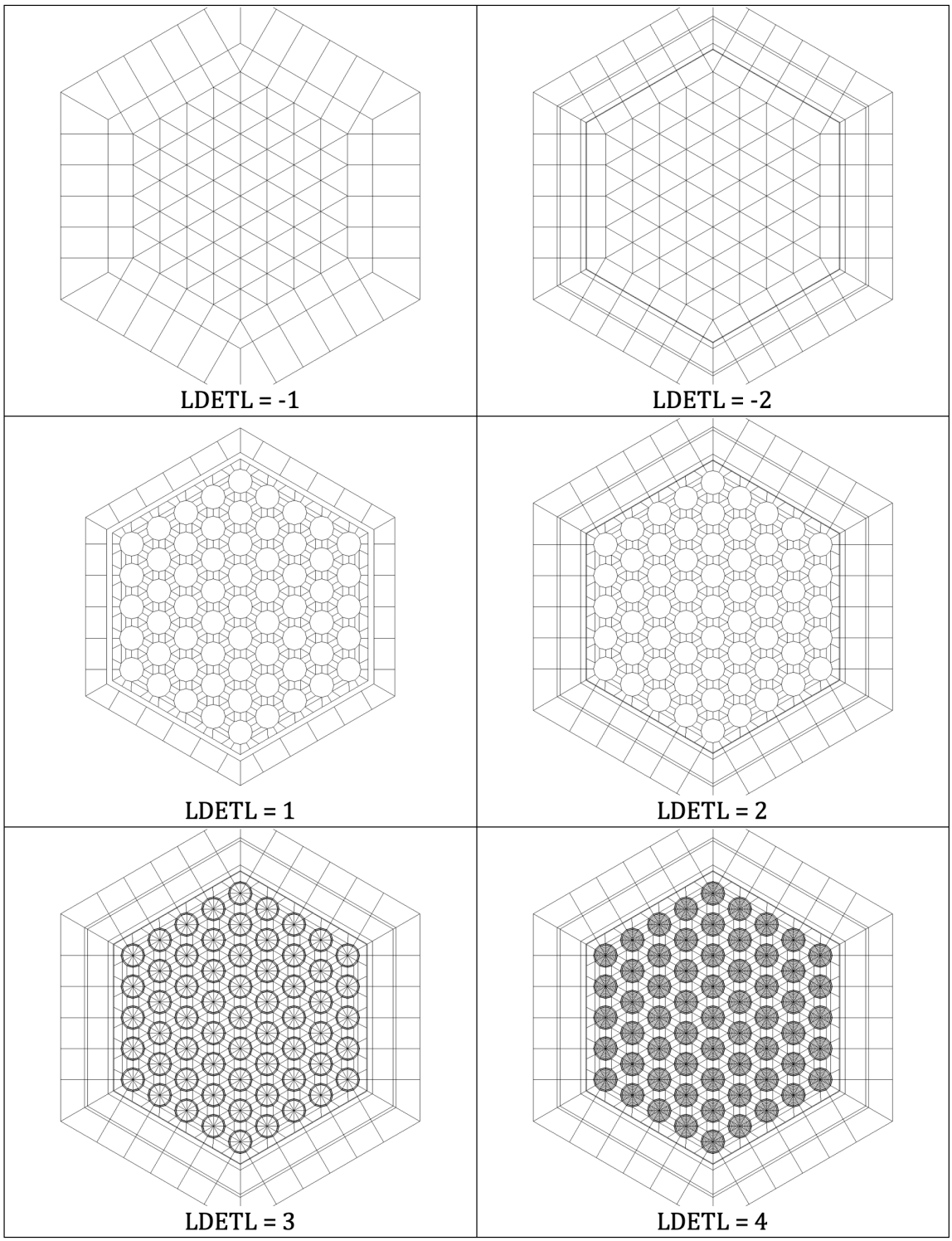


Figure 3.19.11: Examples of Geometry Detail Options Applied to the XX09 Experimental Assembly.

- 1-3: Cladding Radial Components (Inner, Middle, Outer; present when LDETL ≥ 3)
- 4-5: Structure Components (Inner, Outer; present when |LDETL| ≥ 2)
- 10: Bulk Fuel (present when LDETL = 3)
- 11-(10+NT): Radial Fuel Components (present when LDETL = 4)

Data generated during a simulation is associated with a cell in the visualization mesh by matching three parameters: channel ID, axial position, and channel type. This data can then further vary as a function of time. The creation of a SAS4A/SASSYS-1 visualization *data* file is described in the following section.

3.19.3.3 Data Generation

The visualization capabilities introduced into SAS4A/SASSYS-1 assume that geometry is invariant. This significantly simplifies implementation since the SAS4A/SASSYS-1 code has limited knowledge of the specific geometric details needed to project computed simulation data into three-dimensional space. As described in the previous section, a visualization mesh geometry file is created by the multiple-pin subchannel preprocessor. Modifications have been made to the SAS4A/SASSYS-1 code to produce a separate visualization data file that contains time-dependent simulation data that can be associated with the visualization mesh geometry. Details on the format of the visualization data file are contained in Section 3.19.3.7.

Two parameters have been added to the SAS4A/SASSYS-1 input description to control the creation of the visualization data file. These two parameters are defined in Table 3.19.23. The first parameter, IVIS3D, indicates how often to write simulation data to the visualization data file. By default, the file created will be named "SAS.sasdata". Like with the visualization mesh geometry file, the basename of the file is not critical, but the file extension (*.sasdata) is used by the VisIt visualization software to identify the correct file reader with which to parse the data file.

The second parameter, LDETL, is a channel-dependent parameter that controls the extent of data written for each channel. It corresponds to the LDETL parameter defined for the multiple-pin subchannel input preprocessor. When creating channel-dependent input using the input preprocessor, the LDETL parameter will automatically be written to the generated input description. Therefore, a user does not normally need to set this input parameter.

Table 3.19.23: SAS4A/SASSYS-1 Input Parameters for Generation of Visualization Data

Parameter	Description
IVIS3D	Visualization Data Generation Flag: = 0, Do not write visualization data > 0, Write visualization data every IVIS3D time steps.
LDETL	Visualization Data Detail Flag: = 0, Do not write visualization data for this channel = 1, Write coolant temperatures only = 2, Coolant and duct wall temperatures = 3, Coolant, structure, and average fuel temperatures = 4, Coolant, structure, average fuel, and detailed radial fuel temperatures.

There are two minor differences in the treatment of the LDETL parameter between the multiple-pin subchannel input preprocessor and the SAS4A/SASSYS-1 code. The input preprocessor allows negative values for LDETL, which correspond to coolant subchannel cells that completely occupy the assembly dimensions. However, the data needed for these two options is identical to the data needed for positive values of LDETL that have the same magnitude. Therefore, SAS4A/SASSYS-1 only defines positive values for LDETL. The second difference is that for LDETL = 4, SAS4A/SASSYS-1 also writes out the average fuel temperatures (corresponding to LDETL =

3) in addition to the detailed radial fuel temperatures. This is not done for the visualization mesh because it would define multiple cells for the same physical region. By writing this additional data, a visualization data file created as part of a simulation with LDETL = 4 can be used with a visualization mesh geometry file created with LDETL = 3 without having to re-run the simulation. In fact, once a visualization data file is created, it can be combined with a visualization mesh geometry file with a lower-magnitude LDETL parameter.

3.19.3.4 Data Visualization

The preceding sections describe the two-step process needed to create a visualization mesh geometry file and a visualization data file respectively. Interactively viewing the results of a simulation contained in these files is accomplished using the VisIt visualization software package freely available from Lawrence Livermore National Laboratory. VisIt is available for multiple computing platforms, including Linux, Windows, Mac OS X, Solaris, and others. In collaboration with LLNL, a Windows-based file reader plug-in for VisIt has been written that can parse the two files described here and import data into VisIt.

Each of the two files requires a unique file extension (*.sasgeom and *.sasdata) that VisIt uses to identify the correct file-reader plug-in to use when reading the data files. While the geometry file can be viewed independently of the data file, the data file requires a matching geometry file so that data can be projected into three-dimensional space. The matching of the geometry and data files is done by comparing a file's basename. Therefore, SAS.sasdata will be associated with SAS.sasgeom (and vice versa). This process is handled automatically by the file reader.

VisIt supports extensive data visualization capabilities, including pseudo color plots, slices, contours, and animations as well as data picks (queries), filters, time derivatives, and the construction of arbitrary expressions based on existing data. The use of the VisIt visualization tool is beyond the scope of this memo. Details on downloading and using VisIt are currently found at http://www.llnl.gov/visit/.

To illustrate some of the visualization capabilities, two (static) examples from the SHRT-17 simulation are shown in Figure 3.19.12 and Figure 3.19.13. Figure 3.19.12 shows a plot of the XX09 assembly and its neighbors at t = 0 (steady state). The plot has been cropped through the center of XX09 to show the interior detail of fuel-pin temperatures. Figure 3.19.13 shows a contour plot of a slice at the top of the active core region at t = 90 seconds into the transient. Animated versions of both of these plots are presently archived at http://internal.ne.anl.gov/~fanning/SHRT17.html.

3.19.3.5 Summary

An advanced visualization capability has been developed for the SAS4A/SASSYS-1 code that allows for the generation, display, and animation of large datasets that are calculated during a transient simulation. The key features of the new capability include its support for multiple-pin, subchannel modeling, integration with the subchannel input preprocessor, geometry visualization, fully-detailed temperature field support with unlimited dataset sizes, and interactive visualization and animation generation on a desktop PC using the visualization tool VisIt.

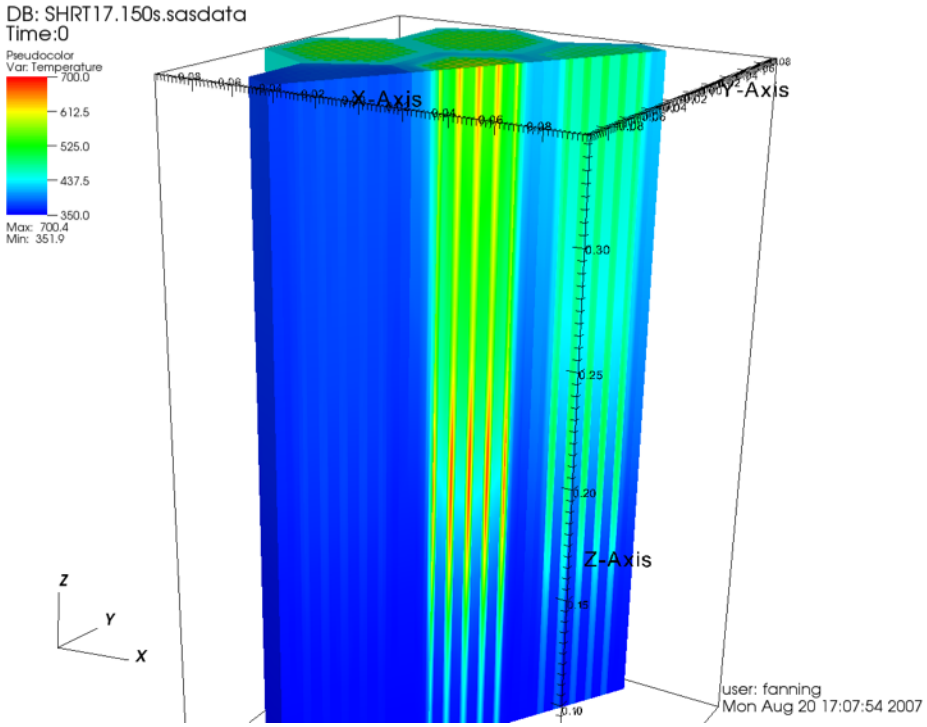


Figure 3.19.12: Pseudo Color Plot of Steady-State Coolant, Structure, and Fuel Temperatures (°C) for XX09 and Surrounding Assemblies.

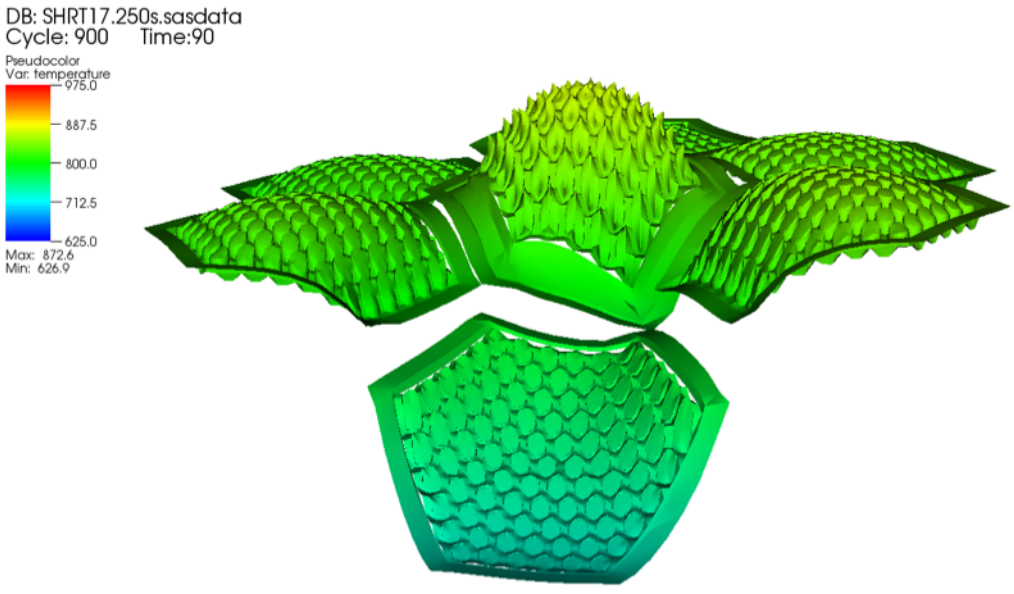


Figure 3.19.13: Contour Plot of Coolant, Structure, and Fuel Temperatures (°K) for XX09 and Adjacent Assemblies. (Top of Core at $t = 90$ seconds)

3.19.3.6 References

1. J. E. Cahalan, et al., "Advanced LMR Safety Analysis Capabilities in the SASSYS-1 and SAS4A Computer Codes," *Proc. Int'l Topical Meeting on Advanced Reactor Safety*, American Nuclear Society, Pittsburgh, PA, April 17-21, 1994.
2. "VisIt User's Manual", UCRL-SM-220449, Lawrence Livermore National Laboratory, October 2005.
3. F. E. Dunn, et al., "Whole Core Subchannel Analysis Verification with the EBR-II SHRT-17 Test," *Proc. ICAPP 2006*, Reno NV, June 4-8, 2006.

3.19.3.7 File Formats

3.19.3.7.1 Visualization Mesh Geometry File

The visualization mesh geometry file is an unformatted (binary) Fortran file that contains a file header followed by two data sections. The first data section defines assembly types in a local coordinate system. The second data section defines actual assemblies in the problem geometry in terms of previously-defined types and offsets into a global coordinate system. This arrangement very much mimics the definition of assembly types and positions in the multiple-pin subchannel preprocessor. All numeric data is represented by four byte integers and double-precision (eight-byte) floating point values. Byte order does not matter, since the file reader for VisIt was designed to detect and support both conventions. However byte order must be consistent within a single file.

3.19.3.7.1.1 File Header

The file header consists of two records that identify file attributes and a problem title. The first record defines the following file attributes:

FILEID File ID. Must be the four characters `SAS_`. (The fourth character is a space.)

FILETYPE File Type. Must be the four characters `GEOM`.

GEOMVER1 Integer File Version. Currently, GEOMVER1 = 1.

GEOMVER2 Integer File Sub-version. Currently, GEOMVER2 = 0.

CHRDATE File Creation Date. (Eight-byte character string.)

CHRTIME File Creation Time. (Eight-byte character string.)

The first four attributes are used by the VisIt file reader to ensure that the correct reader is being used to parse the data file. If major or minor changes to the file format are introduced, the GEOMVER1 and GEOMVER2 attributes will be changed to identify the correct file format for the reader to parse.

The second record in the file header is an 80-character problem title. The multiple-pin subchannel preprocessor uses the problem title provided as input for this record.

3.19.3.7.1.2 Assembly Types

The first data section following the file header defines assembly types in a local coordinate system. Since most assemblies (for example, driver assemblies) will have the same geometry, this simplifies (and shortens) the overall visualization mesh geometry file.

The structure of the assembly type definitions is represented by the following pseudo code:

```
NTYPES

DO ITYPE = 1, NTYPES

  TYPE\_TITLE(ITYPE)

  ITYPEID(ITYPE), NCHAN(ITYPE), NZ(ITYPE),
(Z(K,ITYPE),K=1,NZ(ITYPE))

  DO J = 1, NCHAN(ITYPE)

    IRELCH(J,ITYPE), NP(J,ITYPE),
(X(I,J,ITYPE),Y(I,J,ITYPE),I=1,NP(J,ITYPE))

  ENDDO

ENDDO
```

The assembly type definitions begin with a record containing a single integer parameter, NTYPES, defining the number of assembly types contained in this data section. Each of the assembly type definitions that follow contains a title record (TYPE_TITLE, 80 characters) and an attributes record containing the following parameters:

ITYPEID Unique identification number for this assembly type.

NCHAN Number of "data channels" defined for this assembly type.

NZ Number of axial segments boundaries. (Typically, NZ = MZ + 1.)

Z Z coordinates of the axial segment boundaries in the local coordinate system.

Following the assembly type attributes record is one record for each of the NCHAN data channels defined for an assembly type. The data channel record defines the attributes and geometric shape (in the x-y plane) of the data channel:

IRELCH Relative data channel identification number. (See ICHOFF below.)

NP Number of vertices in the polygon defining this data channel. (Presently, NP must be 3 or 4.)

X, Y (X,Y) coordinates of the vertices defining the polygon for this data channel in local coordinates.

3.19.3.7.1.3 Assembly Definitions

The second data section defines "actual" assemblies in the problem geometry by placing assembly types defined earlier into a global coordinate system. The structure of the assembly definitions is represented by the following pseudo code:

```
NSUBS

DO ISUB = 1, NSUBS

  SUB\_TITLE(ISUB)

  ISUBID(ISUB), ISUBTYPEID(ISUB), ICHOFF(ISUB),
XOFF(ISUB),YOFF(ISUB),ZOFF(ISUB)

ENDDO
```

The subassembly definitions begin with a record containing a single integer parameter, NSUBS, defining the number of assemblies being specified in the visualization mesh geometry file. This need not represent all the assemblies in the problem geometry, only those for which a visualization mesh is needed. Each of the assembly definitions that follow contains a title record (SUB_TITLE, 80 characters) and an assembly attributes record containing the following parameters:

ISUBID Identification number for this subassembly.

ISUBTYPEID Identification number of the assembly type being used to define this assembly.

ICHOFF Integer offset to translate each relative data channel identification number into an absolute (global) data channel identification number.

XOFF, YOFF, ZOFF Coordinate offsets to translate local coordinates of the type definition into global coordinates of the problem geometry.

Absolute data channel identification numbers are constructed from the sum of the relative data channel identification number (IRELCH) and the data channel offset (ICHOFF) for each of the data channels that define an assembly type. For data channels that represent coolant, IRELCH typically starts at 1 and increases by 1 for each coolant subchannel in the assembly type definition. When added to ICHOFF, the resulting data channel identification corresponds to the channel number (ICH) in the SAS4A/SASSYS-1 simulation. Note that because multiple data channels may be used to represent the complex shape of a particular region (the coolant region of a subchannel, for example) absolute data channel identification numbers need not be unique in the visualization mesh geometry file.

For data channels that represent regions other than coolant, an additional term is introduced into the definition of IRELCH. If ICHTYPE represents the channel types as defined by the bulleted list on Page 3-184, then the additional term added to IRELCH is equal to ICHTYPE$\times 10^6$. This procedure is also used by SAS4A/SASSYS-1 to create data channel identification numbers based on the channel number, ICH, and guarantees correct association between the visualization mesh and the simulation data. That is, simulation data is correctly associated with the visualization mesh when (IRELCH + ICHOFF) = (ICHTYPE$\times 10^6$ + ICH).

3.19.3.7.2 Visualization Data File

The visualization data file is an unformatted (binary) Fortran file that contains a file header followed by one data section for each time step saved during a simulation. All numeric data is represented by four byte integers and double-precision (eight-byte) floating point values. Byte order does not matter, since the file reader for VisIt was designed to detect and support both conventions. However byte order must be consistent within a single file.

3.19.3.7.2.1 File Header

The simulation data file header consists of a record that identifies file attributes and two problem title records. The first record defines the following file attributes:

FILEID File ID. Must be the four characters `SAS_`. (The fourth character is a space.)

FILETYPE File Type. Must be the four characters `DATA`.

DATAVER1 Integer File Version. Currently, DATAVER1 = 1.

DATAVER2 Integer File Sub-version. Currently, DATAVER2 = 0.

CHRDATE File Creation Date. (Eight-byte character string.)

CHRTIME File Creation Time. (Eight-byte character string.)

The first four attributes are used by the VisIt file reader to ensure that the correct reader is being used to parse the data file. If major or minor changes to the file format are introduced, the DATAVER1 and DATAVER2 attributes will be changed to identify the correct file format for the reader to parse.

The second and third records in the file header represent two 80-character problem titles. The problem titles are determined by the two title records provided as part of the SAS4A/SASSYS-1 input.

3.19.3.7.2.2 Simulation Data

The remainder of the visualization data file consists of repeated blocks of simulation data. Data blocks contain simulation data for each time step that is recorded as determined by the value of the **IVIS3D** input parameter. The details of the simulation data that is recorded are determined by the channel-dependent input parameter, **LDETL**. The arrangement of data in the simulation data block is described by the following pseudo code:

```
TIME, NDATA

DO J = 1, NDATA

  ICHID(J), ND(J), (DATA(J,K),K=1,ND)

ENDDO
```

The first record in the data block contains the current simulation time (TIME) and the number of data channels that are to follow (NDATA). Although all data blocks from a simulation will have the same length and the same number of data channels, the NDATA parameter is provided to facilitate parsing and verification by file readers.

Following the initial record is a series of NDATA records containing simulation data represented by the following parameters:

ICHID Unique ID for this data channel.

ND Number of axial data values for this data channel. (Typically, ND = MZ.)

DATA Simulation data, with one value for each axial position.

Data channel IDs must be unique within each data block and are associated with zero or more data channels in the visualization mesh geometry in order to project the data into three-dimensional space. The construction of the data channel ID is described in the previous section as ICHID $= \left(\text{ICHTYPE} \times 10^6 + \text{ICH}\right)$ and ensures a unique value based on SAS4A/SASSYS-1 channel numbers (ICH) and the type of data being stored.

3.19.4 APPENDIX 3.4: Coolant Physical Properties

3.19.4.1 Introduction

The expressions used in SAS4A/SASSYS-1 to model coolant physical properties as functions of temperature are derived from data correlations generated by Padilla *[12.1]* and by Fink and Leibowitz *[12-12]*. These correlations were developed using sodium coolant properties that match the most recent measurements available, namely those of Bhise and Bonilla *[12-13]* and Da Gupta and Bonilla *[12-14]*. The mathematical forms of these properties are maintained for coolants independent of their composition.

The mathematical expressions for coolant physical properties are provided in the sections below, including brief descriptions of their origins. In all cases, the temperature T is in Kelvin.

3.19.4.2 Sodium Coolant Physical Properties

Details for sodium coolant properties and the associated polynomial coefficients are described in Section 12.13. This information is not repeated here.

3.19.4.3 Lead Coolant Physical Properties

The lead coolant properties are derived from the data and correlations available in *[A3.4-1]*. The mathematical expressions in *[A3.4-1]* are not directly implemented, but rather a least-squares approximation to these correlations was generated for lead coolant properties to match the existing sodium-derived forms implemented within SAS4A/SASSYS-1. The error introduced by this approach, which is largely considered negligible, has been quantified and documented in the sections below.

3.19.4.3.1 Lead Critical Temperature

The critical temperature (T_c) is an important parameter for the development of an equation of state (EOS) and the extension of information on the available thermophysical data to higher temperatures and pressures. A very large uncertainty exists in the critical temperature for lead as indicated in *[A3.4-1]*.

The literature review performed in *[A3.4-1]* takes into account the large uncertainty in the estimations of the critical temperature. The value recommended is:

$$T_{Crit} = 5000.0 \pm 200 \text{ K} \tag{3.19-61}$$

3.19.4.3.2 Lead Heat of Vaporization

The heat of vaporization at the normal boiling point is also referred to the latent heat (enthalpy) of vaporization in the literature. It quantifies the energy required to break the atomic binding in lead.

The literature review in *[A3.4-1]* outlines a relatively small standard deviation. The recommended value in *[A3.4-1]* for the latent heat of vaporization for pure lead at the normal boiling point is:

$$\mathrm{Q} = 858.6 \pm 1.9 \text{ kJ/kg} \tag{3.19-62}$$

The heat of vaporization correlation implemented in SAS4A/SASSYS-1 uses a temperature-dependent polynomial to compute this quantity:

$$\lambda = A_1 + A_2 T + A_3 T^2 + A_4 T^3 \tag{3.19-63}$$

Lead's latent heat is a constant and, therefore, only A_1 is specified. The corresponding values for the coefficients of Eq. (3.19-63) are provided in Section 3.19.4.6.

3.19.4.3.3 Lead Saturation Pressure

The lead saturation pressure as a function of temperature is an important property of coolants. It determines the dependence of the boiling temperature on pressure and heat-mass transfer in the cover gas space. It is directly related to the latent heat (enthalpy) of vaporization. *[A3.4-1]* provides two correlations for the saturation pressure. The expression chosen for the latent heat of vaporization for pure lead at the normal boiling point is:

$$P_S = 1.88\text{×}10^{13} T^{-0.985} e^{\left(\frac{-23325}{T}\right)} \tag{3.19-64}$$

The saturated vapor pressure is defined within a temperature range of $[\mathrm{T}_{M,0(Pb)} - \mathrm{T}_{B,0(Pb)}]$ where $\mathrm{T}_{M,0(Pb)}$ represents the normal melting point (600.6 K) and $\mathrm{T}_{B,0(Pb)}$ is the normal boiling point $(2021 \pm 3\text{ K})$. Per *[A3.4-1]*, the standard deviation with respect to the data is:

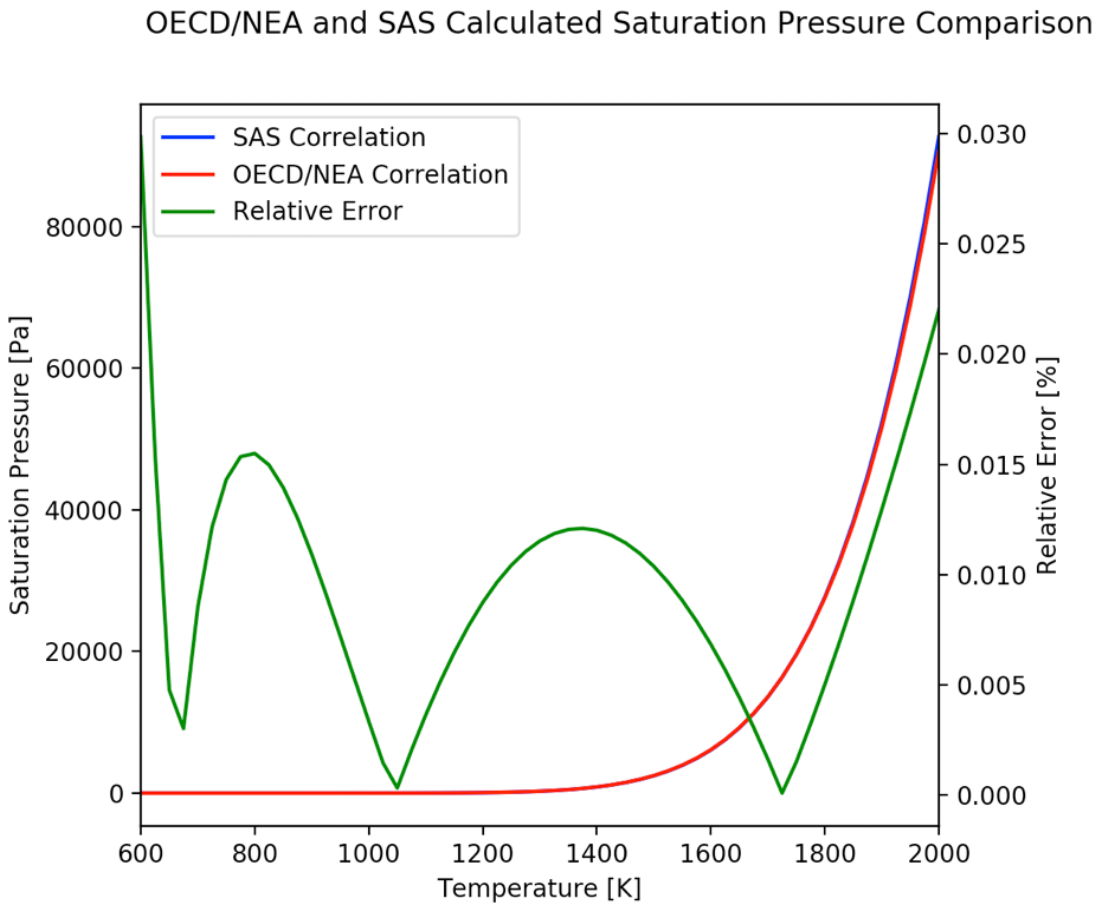
- 15% for temperatures within the range $[\mathrm{T}_{M,0(Pb)} - 1100\text{ K}]$
- 5% for temperatures within the range $[1100\text{ K} - \mathrm{T}_{B,0(Pb)}]$

The calculation of the saturation pressure in SAS4A/SASSYS-1 correlates the natural logarithm of the saturation pressure to a temperature-dependent polynomial.

$$ln(P_S) = A_5 - \frac{A_6}{T} - \frac{A_7}{T^2} \tag{3.19-65}$$

The constant coefficients in the SAS4A/SASSYS-1 second order correlation were evaluated using a least squares equal weighted residual minimization fit to the OECD compressibility correlation. The corresponding values for the coefficients of Eq. (3.19-65) are provided in Section 3.19.4.6.

The temperature range provided in *[A3.4-1]* was maintained when performing the least square fit to adapt to the form of Eq. (3.19-65). Uncertainties for this correlation are not determined. However, absolute and relative errors with respect to the OECD correlation were calculated. The relative error is provided in the plot below.


OECD/NEA and SAS Calculated Saturation Pressure Comparison
SAS Correlation
OECD/NEA Correlation
Relative Error
Saturation Pressure [Pa]
Relative Error [%]
Temperature [K]
0
20000
40000
60000
80000
0.000
0.005
0.010
0.015
0.020
0.025
0.030
600
800
1000
1200
1400
1600
1800
2000

3.19.4.3.4 Lead Saturation Temperature

SAS4A/SASSYS-1 requires an expression for temperature as a function of pressure. This expression is obtained by inverting Eq. (3.19-65). Direct inversion yields:

$$T_S = \frac{A_8}{A_9 + \sqrt{A_{10} + A_{11} ln\left(P_S\right)}} \tag{3.19-66}$$

The corresponding values for the coefficients of Eq. (3.19-66) are provided in Section 3.19.4.6.

3.19.4.3.5 Lead Liquid Density

The lead density depends on temperature and pressure. The expression recommended in *[A3.4-1]* for the lead density defines a linear temperature dependence of the form:

$$\rho_l = 11441 - 1.2795T \tag{3.19-67}$$

The accuracy with which this correlation reproduces selected literature sources is discussed in *[A3.4-1]*. The maximum difference does not exceed 1% within the temperature range from the normal melting point $\mathrm{T}_{M,0(Pb)}$ to 1500 K. This difference becomes about 2% when temperature increases up to the normal boiling point (defined as $\mathrm{T}_{B,0(Pb)} = 2021 \pm 3$ K).

The SAS4A/SASSYS-1 mathematical expression correlating the coolant density to the temperature is a second order polynomial:

$$\rho_l = A_{12} + A_{13}T + A_{14}T^2 \tag{3.19-68}$$

The corresponding values for the coefficients of Eq. (3.19-68) are provided in Section 3.19.4.6. The correlation accuracy and temperature range of *[A3.4-1]* is consistent with Eq. (3.19-68) since the correlation mathematical form is maintained.

3.19.4.3.6 Lead Vapor Density

Discussion of lead vapor density is not available in *[A3.4-1]*, and alternative literature/experimental sources are not widely available.

The calculation of the vapor density in SAS4A/SASSYS-1 uses a temperature-dependent polynomial which is multiplied by the saturation temperature:

$$\rho_v = P_S \left(\frac{A_{15}}{T} + A_{16} + A_{17}T + A_{18}T^2 + A_{19}T^3 + A_{20}T^4 \right) \tag{3.19-69}$$

The corresponding values for the coefficients of Eq. (3.19-69) are provided in Section 3.19.4.6. For the current implementation, a value for $A_{15} = 0.01$ is assumed, and all other coefficients are assumed to be zero. The temperature range over which this correlation is applicable is unknown, but it is assumed that it applies to temperatures above $\mathrm{T}_{B,0(Pb)} = 2021 \pm 3$ K where lead vapor is anticipated to be present.

3.19.4.3.7 Lead Liquid Specific Heat

Calculation of the coolant enthalpy change as a function of temperature requires knowledge of the isobaric heat capacity. To account for large uncertainties in the data, *[A3.4-1]* proposes the following expression for liquid lead specific heat:

$$C_l = 176.2 - 4.923 \times 10^{-2} T + 1.544 \times 10^{-5} T^2 - \frac{1.524 \times 10^6}{T^2} \tag{3.19-70}$$

[A3.4-1] identifies a 1991 correlation, which was later refined, as a reasonable choice for the isobaric heat capacity of pure liquid lead in the temperature range of $[\mathrm{T}_{B,0(Pb)} - 2000\ \mathrm{K}]$. The accuracy with which this refined correlation reproduces selected literature sources is discussed in *[A3.4-1]*. The estimated uncertainty is 5-7% within $[\mathrm{T}_{B,0(Pb)} - 1500\ \mathrm{K}]$ and 7-15% at higher temperatures where no experimental data is available in literature.

In SAS4A/SASSYS-1, the liquid specific heat is approximated by a polynomial that considers the difference between coolant temperature and critical temperature Eq. (3.19-61):

$$C_l = \frac{A_{28}}{(T_C - T)^2} + \frac{A_{29}}{(T_C - T)} + A_{30} + A_{31}(T_C - T) + A_{32}\,(T_C - T)^2 \tag{3.19-71}$$

The mathematical expressions in *[A3.4-1]* and that of SAS4A/SASSYS-1 have different forms. The constant coefficients in the SAS4A/SASSYS-1 correlation were evaluated using a least squares equal weighted residual minimization fit to the *[A3.4-1]* specific heat correlation. The temperature range over which this correlation was determined in *[A3.4-1]* is maintained as the least square fit to Eq. (3.19-71) was applied over the range outlined above. The corresponding values for the coefficients of Eq. (3.19-71) are provided in Section 3.19.4.6.

Absolute and relative errors with respect to the OECD correlation were calculated, with relative error provided in the plot below.

3.19.4.3.8 Lead Vapor Specific Heat

Reference *[A3.4-1]* does not discuss vapor specific heat. No experimental data related to the specific heat of lead vapor could be found.

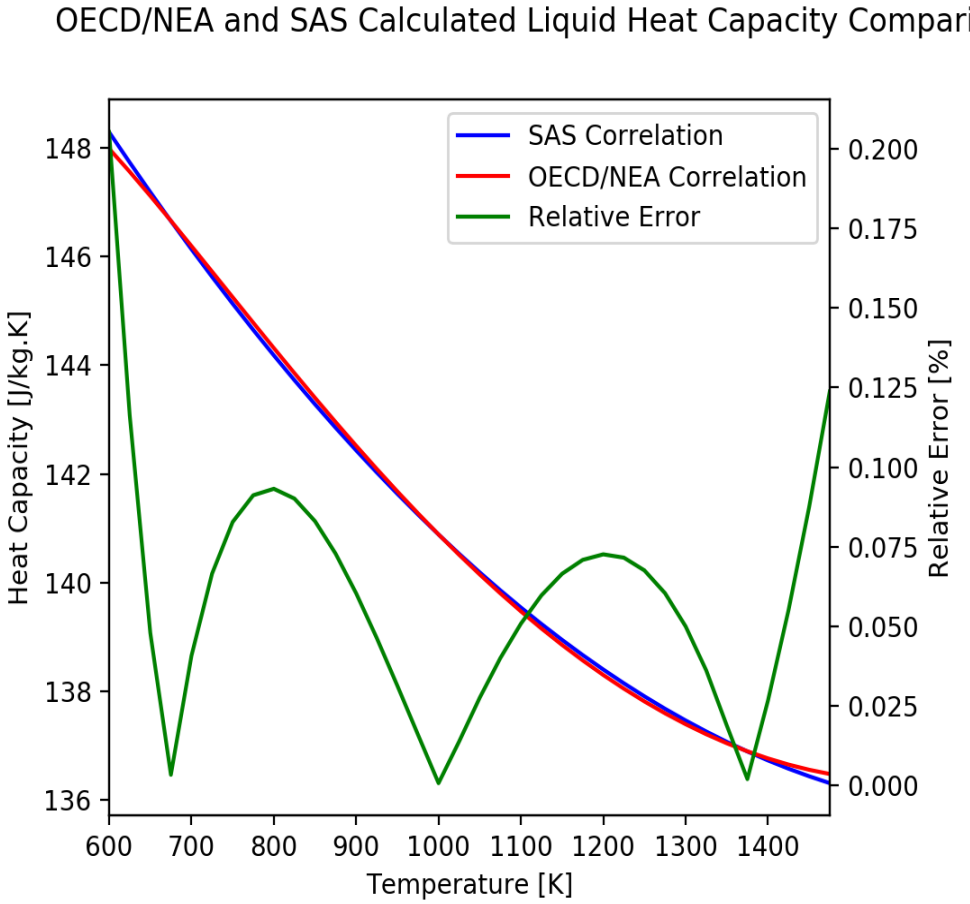
Lead boiling is not expected to occur within the temperature ranges typically considered for lead fast reactors in the design basis analysis envelope. Moreover, in the unlikely event that lead boiling were to occur, this phenomenon will likely be localized for these particular scenarios. Due to the lack of literature on this subject, its limited relevance the typical SAS4A/SASSYS-1 analysis scope, and unavailability of a two-phase lead coolant model in SAS4A/SASSYS-1, vapor specific heat is ignored. The corresponding values for the related coefficients are provided in Section 3.19.4.6.

It should be noted that this assumption is not applicable for severe accidents where coolant boiling occurs at a larger scale.

3.19.4.3.9 Lead Adiabatic Compressibility

Reference *[A3.4-1]* assumes compressibility is a reversible adiabatic process. As such, the isentropic compressibility of the coolant at normal pressure is calculated as the inverse of the bulk elastic modulus. The bulk elastic modulus discussed in *[A3.4-1]* uses a polynomial as a function of temperature. As such, the expression for the compressibility becomes:

$$\beta_s = \left(\left(43.50 - 1.552 \times 10^{-2} T + 1.622 \times 10^{-6} T^2\right) \times 10^9\right)^{-1} \tag{3.19-72}$$


OECD/NEA and SAS Calculated Liquid Heat Capacity Comparison
SAS Correlation
OECD/NEA Correlation
Relative Error
Heat Capacity [J/kg.K]
Relative Error [%]
Temperature [K]
148
146
144
142
140
138
136
0.200
0.175
0.150
0.125
0.100
0.075
0.050
0.025
0.000
600
700
800
900
1000
1100
1200
1300
1400

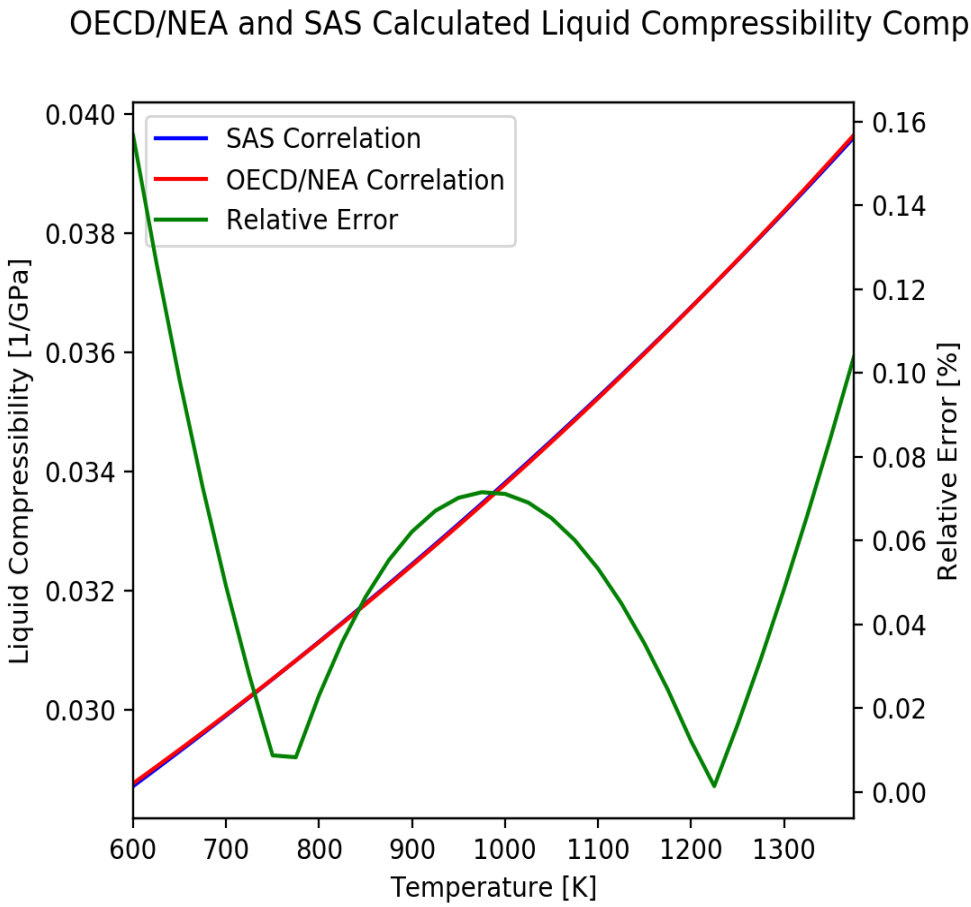
The accuracy with which this correlation predicts compressibility is not directly discussed in *[A3.4-1]*. The report outlines how both compressibility and sound velocity are basic thermodynamic properties of molten lead. The Dreyfus thermodynamic relationship is used to infer the sound celerity from the bulk elastic modulus, and compressibility. Per *[A3.4-1]*, the correlation is applicable for sound celerity of the longitudinal sound waves in molten lead for temperatures up to 1400K (i.e. $[\mathrm{T}_{M,0(Pb)} - 1400\ \mathrm{K}]$). This range is assumed to be the same for compressibility.

The compressibility coefficient is calculated in SAS4A/SASSYS-1 by a polynomial expression based on the inverse of the coolant temperature subtracted from the critical temperature Eq. (3.19-61).

$$\beta_s = A_{40} + \frac{A_{41}}{(T_C - T)} \tag{3.19-73}$$

The mathematical expressions in *[A3.4-1]* and SAS4A/SASSYS-1 have different forms. The constant coefficients in the SAS4A/SASSYS-1 correlation were evaluated using a least squares equal weighted residual minimization fit to Eq. (3.19-72). The corresponding values for the coefficients of Eq. (3.19-73) are provided in Section 3.19.4.6.

Uncertainties for this correlation were not determined. However, absolute and relative errors with respect to Eq. (3.19-72) were calculated. The relative error is provided in the plot below.



3.19.4.3.10 Lead Thermal Expansion

The density of liquid metals changes with temperature due to thermal expansion related to anharmonicity of interatomic forces. As outlined in *[A3.4-1]*, the relationship between the density and the coefficient of volumetric thermal expansion (CVTE) at constant pressure p is defined by the following formula:

$$\alpha_p = -\frac{1}{\rho}\left(\frac{\partial \rho}{\partial T}\right)_p \tag{3.19-74}$$

Substituting the correlations for density recommended by *[A3.4-1]* yields:

$$\alpha_p = \frac{1}{8942 - T} \tag{3.19-75}$$

Since the thermal expansion correlation is defined from that of the density, the same temperature range of $[\mathrm{T}_{M,0(Pb)} - 1400\ \mathrm{K}]$ applies. The uncertainty of this quantity can be derived from the thermal expansion expression and the coolant density uncertainty.

The thermal expansion coefficient is calculated in SAS4A/SASSYS-1 by a polynomial expression based on the inverse of the coolant temperature subtracted to the critical temperature:

$$\alpha_p = A_{42} + \frac{A_{43}}{(T_C - T)} + \frac{A_{44}}{(T_C - T)^2} + \frac{A_{45}}{(T_C - T)^3} + \frac{A_{46}}{(T_C - T)^4} + \frac{A_{47}}{(T_C - T)^5} \tag{3.19-76}$$

A simple polynomial form matching between equations in Eq. (3.19-75) and Eq. (3.19-76) does not allow the SAS4A/SASSYS-1 expression to match that of *[A3.4-1]* with a reasonably small variation. Therefore, additional terms had to be included in the SAS4A/SASSYS-1 correlation. The correlation implemented includes terms from Eq. (3.19-76) up to the third order. The constant coefficients in the SAS4A/SASSYS-1 correlation were evaluated using a least squares equal weighted residual minimization fit to the OECD thermal expansion correlation. The corresponding values for the coefficients of Eq. (3.19-76) are provided in Section 3.19.4.6.

Uncertainties for this correlation were not determined. However, absolute and relative errors with respect to Eq. (3.19-75) were calculated. The relative error is provided in the plot below.

3.19.4.3.11 Lead Thermal Conductivity

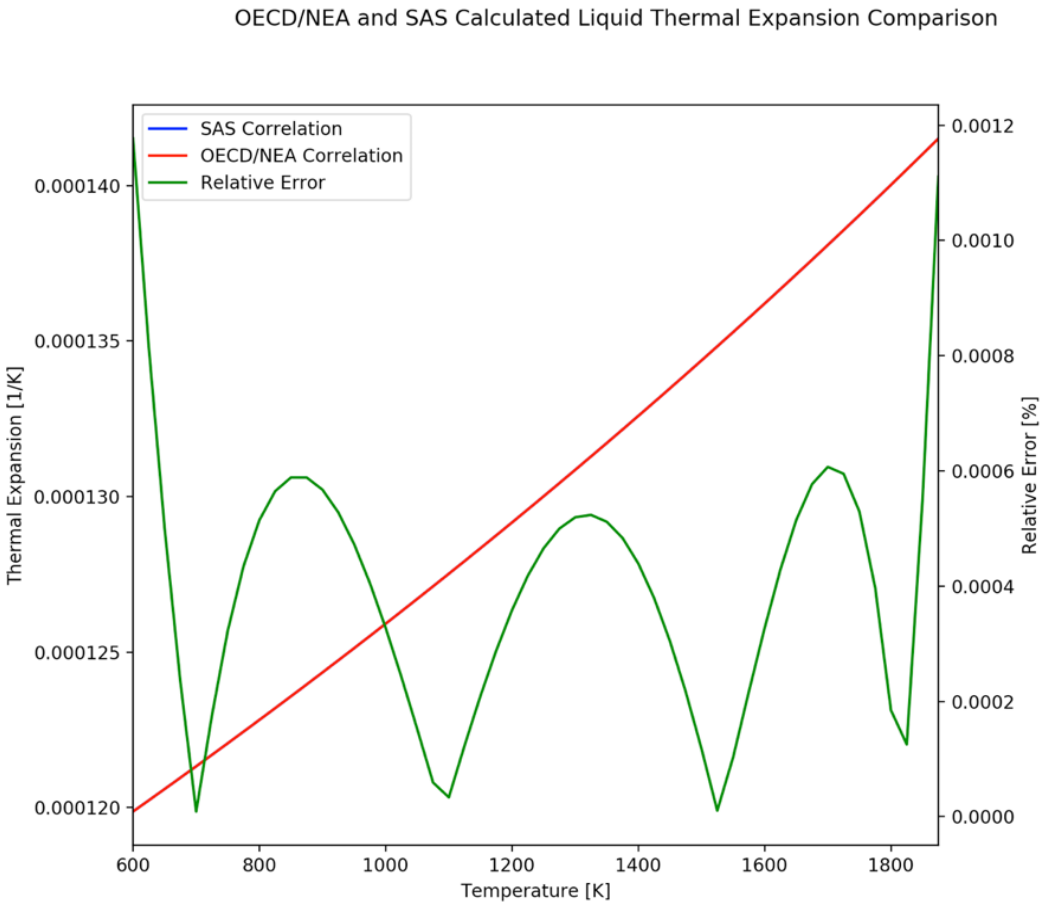
Thermal conductivity is the fundamental thermophysical parameter of liquid metal coolants, which is used in all heat transfer calculations. The accuracy of its determination for liquid metals is poor and usually large discrepancies exist among different sets of data. Per *[A3.4-1]*, the following linear correlation was used as the thermal conductivity of the technically pure liquid lead at normal pressure:

$$k_l = 9.2 + 0.011T \tag{3.19-77}$$

The accuracy with which this correlation reproduces selected literature sources is discussed in *[A3.4-1]*. The correlation is applicable from the lead melting temperature at normal atmospheric pressure up to 1300K (i.e. $[\mathrm{T}_{M,0(Pb)} - 1300\ \mathrm{K}]$) with an estimated standard deviation of 15%. The SAS4A/SASSYS-1 mathematical expression correlating the coolant density to the temperature is a third order polynomial as defined below:

$$k_l = A_{48} + A_{49}T + A_{50}T^2 + A_{51}T^3 \tag{3.19-78}$$

The corresponding values for the coefficients of Eq. (3.19-78) are provided in Section 3.19.4.6. The correlation accuracy and temperature ranges of Eq. (3.19-77) and Eq. (3.19-78) are consistent since the mathematical correlation form is maintained.


OECD/NEA and SAS Calculated Liquid Thermal Expansion Comparison
SAS Correlation
OECD/NEA Correlation
Relative Error
Thermal Expansion [1/K]
Relative Error [%]
Temperature [K]
0.000140
0.000135
0.000130
0.000125
0.000120
0.0012
0.0010
0.0008
0.0006
0.0004
0.0002
0.0000
600
800
1000
1200
1400
1600
1800

3.19.4.3.12 Lead Viscosity

Viscosity is a measure of friction among atoms of liquids. Because there is very little data on the impact of impurities on viscosity, the correlations below are for pure lead. Lead is believed to be a Newtonian liquid. Therefore, the temperature dependence of their dynamic viscosity is usually described by an Arrhenius-type equation as described in *[A3.4-1]*.

$$\mu_l = 4.55 \times 10^{-4} e^{\left(\frac{1069}{T}\right)} \tag{3.19-79}$$

This correlation is valid in the temperature range from $T_{\text{M,0(Pb)}}$ to 1473K with an estimated uncertainty of $\pm 5\%$. Coolant dynamic viscosity is calculated in SAS4A/SASSYS-1 by a polynomial expression based on the inverse of the coolant temperature.

$$\mu_l = A_{52} + \frac{A_{53}}{T} + \frac{A_{54}}{T^2} + \frac{A_{55}}{T^3} \tag{3.19-80}$$

The mathematical expressions in Eq. (3.19-79) and Eq. (3.19-80) have different forms. The constant coefficients in the SAS4A/SASSYS-1 correlation were evaluated using a least squares equal weighted residual minimization fit to the OECD viscosity correlation. The corresponding values for the coefficients of Eq. (3.19-80) are provided in Section 3.19.4.6.

The temperature range over which this correlation was determined in *[A3.4-1]* is maintained because the least square fit to Eq. (3.19-79) was performed over the applicability range outlined above. Uncertainties for this correlation were not determined. However, absolute and relative errors with respect to Eq. (3.19-79) were calculated. The relative error is provided in the plot below.

3.19.4.4 LBE Coolant Physical Properties

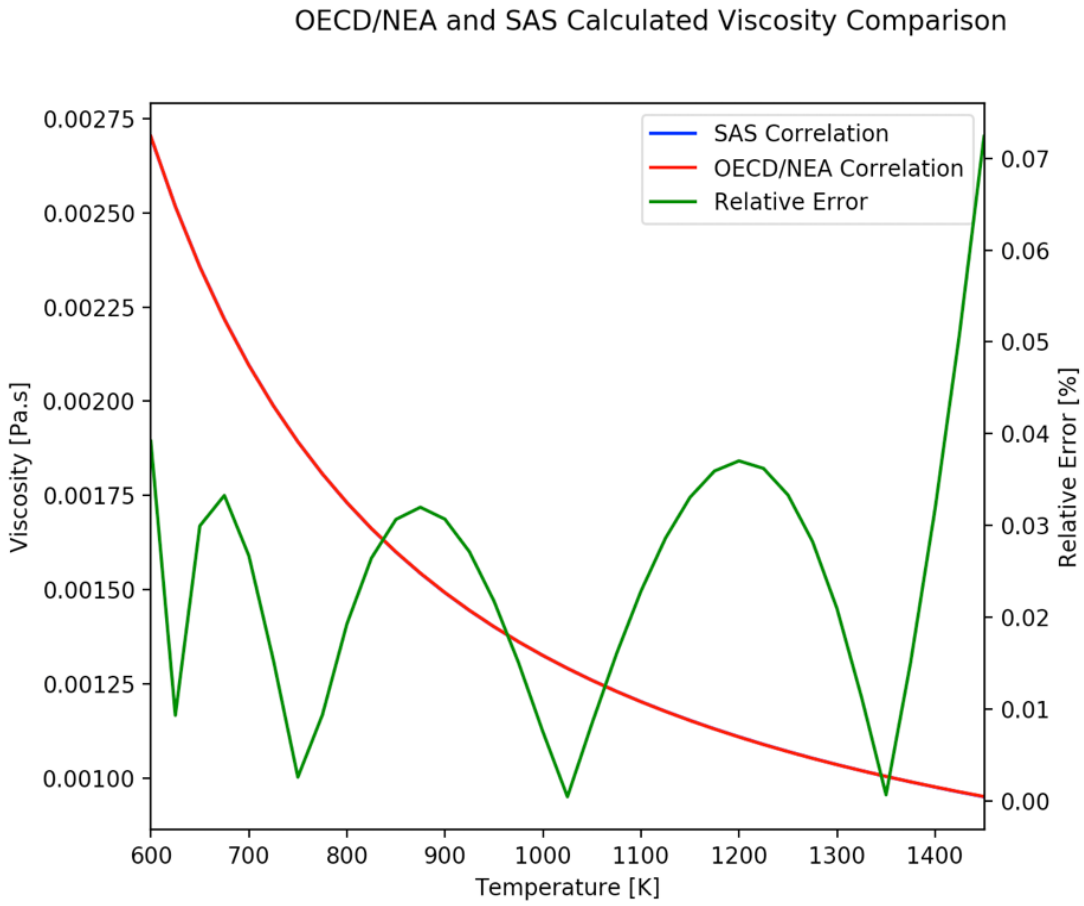
The LBE coolant properties are derived from the data and correlations available in *[A3.4-1]*. The mathematical expressions in *[A3.4-1]* are not directly implemented for some properties, but rather a least-squares approximation to these correlations was generated for coolant properties to match the existing sodium-derived forms implemented within SAS4A/SASSYS-1. The error introduced by this approach, which is largely considered negligible, has been quantified and documented in the sections below.

3.19.4.4.1 LBE Critical Temperature

The critical temperature (T_c) is an important parameter for the development of an equation of state (EOS) and the extension of information on the available thermophysical data to higher temperatures and pressures.

A very large uncertainty still exists in the critical temperature for LBE as indicated in *[A3.4-1]*, as no measurement data is available in public literature. Critical parameters are only available through theoretical estimations. The literature review performed in *[A3.4-1]* takes into account the three available estimations of the critical temperature. The value recommended by *[A3.4-1]*, bounded by the uncertainties of lead and bismuth, is:

$$\mathrm{T}_{Crit} = 4800.0 \pm 500 \text{ K} \tag{3.19-81}$$


OECD/NEA and SAS Calculated Viscosity Comparison
SAS Correlation
OECD/NEA Correlation
Relative Error
Viscosity [Pa.s]
Relative Error [%]
Temperature [K]
0.00275
0.00250
0.00225
0.00200
0.00175
0.00150
0.00125
0.00100
0.07
0.06
0.05
0.04
0.03
0.02
0.01
0.00
600
700
800
900
1000
1100
1200
1300
1400

3.19.4.4.2 LBE Heat of Vaporization

The heat of vaporization at the normal boiling point is also referred to the latent heat (enthalpy) of vaporization in the literature. It quantifies the energy required to break the atomic binding in LBE. The literature review in *[A3.4-1]* indicates limited availability of information. The recommended value in *[A3.4-1]* for the latent heat of vaporization of pure LBE at the normal boiling point is:

$$\mathrm{Q} = 856 \pm 5 \text{ kJ/kg} \tag{3.19-82}$$

The heat of vaporization correlation implemented in SAS4A/SASSYS-1 uses a temperature-dependent polynomial to compute this quantity:

$$\lambda = A_1 + A_2 T + A_3 T^2 + A_4 T^3 \tag{3.19-83}$$

The latent heat of LBE is a constant and, therefore, only A_1 is specified. The corresponding values for the coefficients of Eq. (3.19-83) are provided in Section 3.19.4.6.

3.19.4.4.3 LBE Saturation Pressure

Vapor pressure as a function of temperature is an important property of coolants. It determines the dependence of the boiling temperature on pressure and heat-mass transfer in the cover gas space. It is directly related to the latent heat (enthalpy) of vaporization. The expression recommended by *[A3.4-1]* for the saturation pressure for pure LBE is:

$$P_S = 1.22 \times 10^{10} e^{\left(\frac{-22552}{T}\right)} \tag{3.19-84}$$

The literature review in *[A3.4-1]* indicates very large uncertainties exist at low temperatures $(< 773\mathrm{K})$. Experimental data used to develop the above correlation is defined within a temperature range of $[500 - 1943\mathrm{K}]$. The OECD correlation standard deviation with respect to the data is, per *[A3.4-1]*:

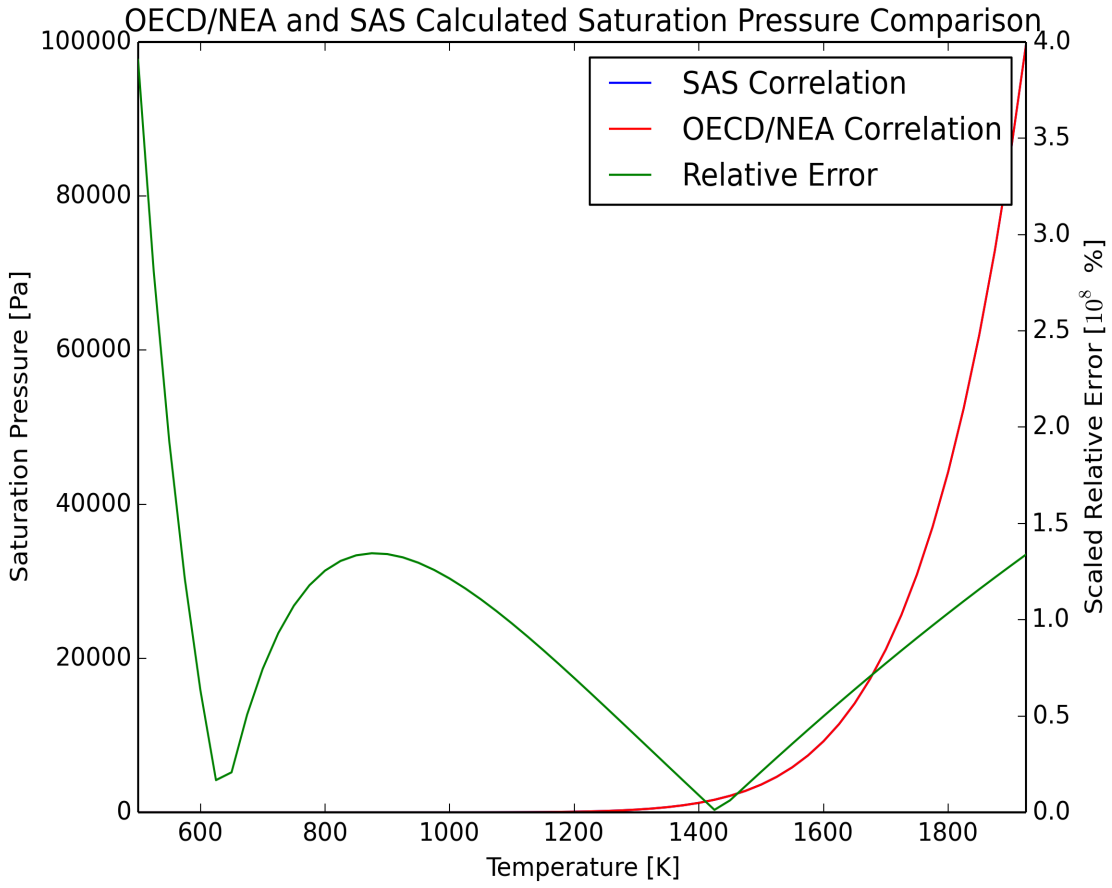
- $\pm 60\%$ for temperatures within the range $[673 - 1927\mathrm{K}]$, and
- No agreement below 673 K.

The calculation of the saturation pressure in SAS4A/SASSYS-1 correlates the natural logarithm of the saturation pressure to a temperature-dependent polynomial.

$$ln(P_S) = A_5 - \frac{A_6}{T} - \frac{A_7}{T^2} \tag{3.19-85}$$

The constant coefficients in the SAS4A/SASSYS-1 second order correlation were evaluated using a least squares equal weighted residual minimization fit to the OECD correlation. The corresponding values for the coefficients for saturation pressure are provided in Section 3.19.4.6.

The temperature range provided in *[A3.4-1]* was maintained when performing the least square fit to adapt to the form of Eq. (3.19-85). Uncertainties for this correlation are not determined. However, absolute and relative errors with respect to the OECD correlation were calculated. The relative error is provided in the plot below.


OECD/NEA and SAS Calculated Saturation Pressure Comparison
SAS Correlation
OECD/NEA Correlation
Relative Error
Saturation Pressure [Pa]
Scaled Relative Error [10^8 %]
Temperature [K]
100000
80000
60000
40000
20000
0
4.0
3.5
3.0
2.5
2.0
1.5
1.0
0.5
0.0
600
800
1000
1200
1400
1600
1800

3.19.4.4.4 LBE Saturation Temperature

SAS4A/SASSYS-1 requires an expression for temperature as a function of pressure. This expression is obtained by inverting Eq. (3.19-85). Direct inversion yields:

$$T_S = \frac{A_8}{A_9 + \sqrt{A_{10} + A_{11} ln\left(P_S\right)}} \tag{3.19-86}$$

The corresponding values for the coefficients of Eq. (3.19-86) are provided in Section 3.19.4.6.

3.19.4.4.5 LBE Liquid Density

The LBE density depends on temperature and pressure. The expression recommended in *[A3.4-1]* for the lead density defines a linear temperature dependence of the form:

The coolant density depends on temperature and pressure. This quantity is essential to the development of an EOS. The expression recommended for the LBE density defines a linear temperature dependence of the form *[A3.4-1]*:

$$\rho_l = 11065 - 1.293T \tag{3.19-87}$$

The accuracy with which this correlation reproduces selected literature sources is discussed in *[A3.4-1]*. The maximum difference does not exceed 0.8% within the temperature range of $[673 - 1927\text{K}]$.

The SAS4A/SASSYS-1 mathematical expression correlating the coolant density to the temperature is a second order polynomial:

$$\rho_l = A_{12} + A_{13}T + A_{14}T^2 \tag{3.19-88}$$

The corresponding values for the coefficients of Eq. (3.19-88) are provided in Section 3.19.4.6. The correlation accuracy and temperature range of *[A3.4-1]* is consistent with Eq. (3.19-88) since the correlation mathematical form is maintained.

3.19.4.4.6 LBE Vapor Density

Reference *[A3.4-1]* does not discuss vapor density. No experimental data related to this property could be found. Boiling is not expected to occur within the temperature range anticipated for design basis accidents in LBE-cooled fast reactors. However, in the unlikely event that coolant boiling were to occur, this phenomenon will likely be localized. Because of degraded heat transfer to vapor form coolant, coolant vapor temperature is not expected to raise significantly above saturation temperature.

The calculation of the vapor density in SAS4A/SASSYS-1 uses a temperature-dependent polynomial which is multiplied by the saturation pressure:

$$\rho_v = P_S \left(\frac{A_{15}}{T} + A_{16} + A_{17}T + A_{18}T^2 + A_{19}T^3 + A_{20}T^4 \right) \tag{3.19-89}$$

The corresponding values for the coefficients of Eq. (3.19-89) are provided in Section 3.19.4.6. For the current implementation, a value for $A_{15} = 0.01$ is assumed, and all other coefficients are assumed to be zero. The temperature range over which this correlation is applicable is unknown, but it is assumed that it applies to temperatures above $\text{T}_{B,0(LBE)} = 1927 \pm 16$ K where LBE vapor is anticipated to be present.

3.19.4.4.7 LBE Liquid Specific Heat

Heat capacity is one of the most important properties of a coolant. Calculation of the coolant enthalpy change as a function of temperature requires knowledge of the isobaric heat capacity.

Discussion in *[A3.4-1]* indicates a limited availability of independent data on liquid specific heat. Leveraging the similarity of Pb-Bi to its components, available experimental data, and Kopp's law (typically used to derive heat capacities of binary systems), a single polynomial was developed in *[A3.4-1]*. As such, the expression for the molten coolant specific heat becomes:

$$C_l = 164.8 - 3.94 \times 10^{-2} T + 1.25 \times 10^{-5} T^2 - \frac{4.56 \times 10^5}{T^2} \tag{3.19-90}$$

The OECD correlation uncertainty with respect to the data is, per *[A3.4-1]*:

- < 5% for temperatures within the range [400–1100K], and
- 1 – 2% within the range $[1100 - \mathrm{T}_{B,0(LBE)} = 1927 \pm 16\mathrm{K}$, where the validity of Kopp's law is most effective.

In SAS4A/SASSYS-1, the liquid specific heat is approximated by a polynomial that considers the difference between coolant temperature and critical temperature Eq. (3.19-61):

$$C_l = \frac{A_{28}}{(T_C - T)^2} + \frac{A_{29}}{(T_C - T)} + A_{30} + A_{31}(T_C - T) + A_{32}\,(T_C - T)^2 \tag{3.19-91}$$

The mathematical expressions in *[A3.4-1]* and that of SAS4A/SASSYS-1 have different forms. The constant coefficients in the SAS4A/SASSYS-1 correlation were evaluated using a least squares equal weighted residual minimization fit to the *[A3.4-1]* specific heat correlation. The temperature range over which this correlation was determined in *[A3.4-1]* is maintained as the least square fit to Eq. (3.19-91) was applied over the range outlined above. The corresponding values for the coefficients of Eq. (3.19-91) are provided in Section 3.19.4.6.

Absolute and relative errors with respect to the OECD correlation were calculated, with relative error provided in the plot below.

3.19.4.4.8 LBE Vapor Specific Heat

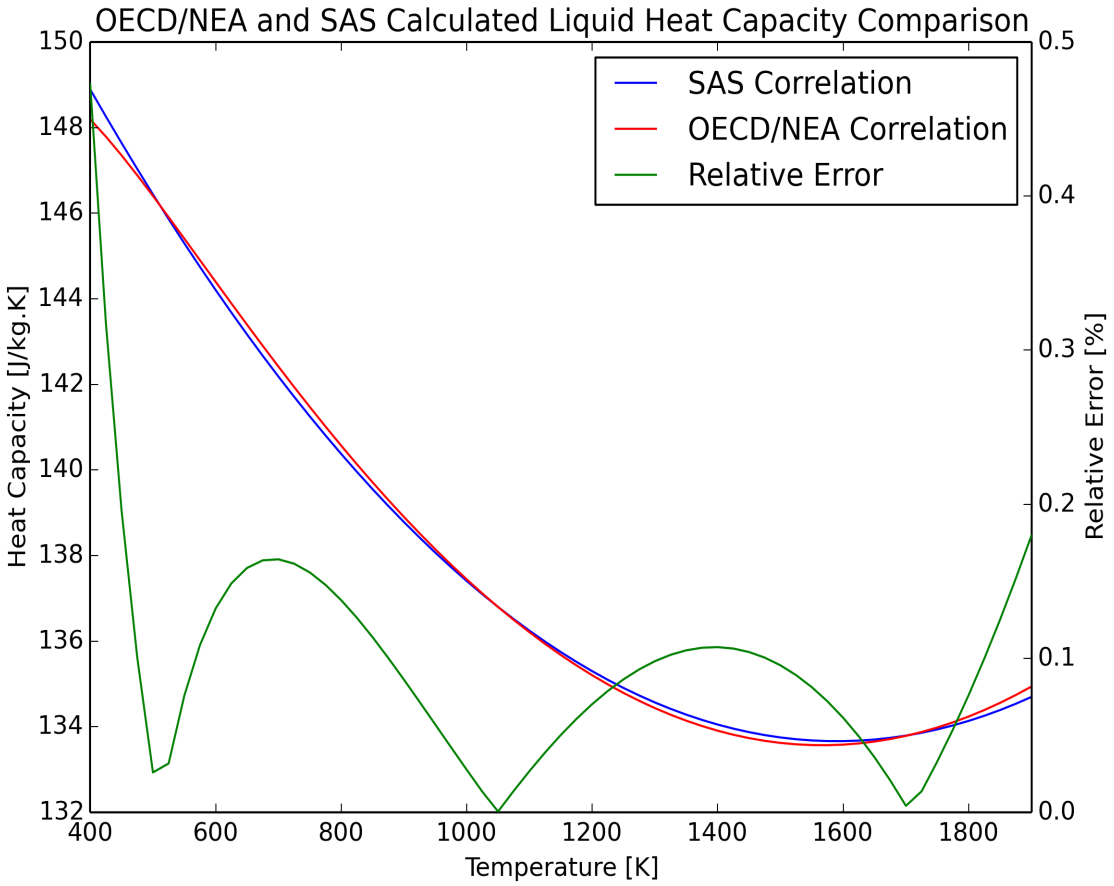
Reference *[A3.4-1]* does not discuss vapor specific heat. No experimental data related to this property could be found. Boiling is not expected to occur within the temperature range anticipated for design basis accidents in LBE-cooled fast reactors. However, in the unlikely event that coolant boiling were to occur, this phenomenon will likely be localized. Because of degraded heat transfer to vapor form coolant, coolant vapor temperature is not expected to raise significantly above saturation temperature.

Specific heat is ignored until more information becomes available. This is not applicable for severe accidents where coolant boiling occurs at a larger scale and for higher temperatures.

3.19.4.4.9 LBE Adiabatic Compressibility

No direct measurement of compressibility are available in literature, as per the discussion in *[A3.4-1]*. Reference *[A3.4-1]* assumes compressibility in this case as a reversible adiabatic process. As such, the isentropic compressibility of the coolant at normal pressure is calculated as the inverse of the bulk elastic modulus. Compressibility is then found from direct measurements of sound velocity, u_s, and density:

$$\beta_s = \left(\rho u_s^2\right)^{-1}$$


OECD/NEA and SAS Calculated Liquid Heat Capacity Comparison
SAS Correlation
OECD/NEA Correlation
Relative Error
Heat Capacity [J/kg.K]
Relative Error [%]
Temperature [K]
150
148
146
144
142
140
138
136
134
132
0.5
0.4
0.3
0.2
0.1
0.0
400
600
800
1000
1200
1400
1600
1800

The bulk elastic modulus discussed in *[A3.4-1]* uses a polynomial as a function of temperature. As such, the expression for the compressibility becomes:

$$\beta_s = \left(3.8075 \times 10^{10} - 1.3152 \times 10^{7} T + 1.5143 \times 10^{3} T^2 - 5.8113 \times 10^{-2} T^3\right)^{-1} \tag{3.19-92}$$

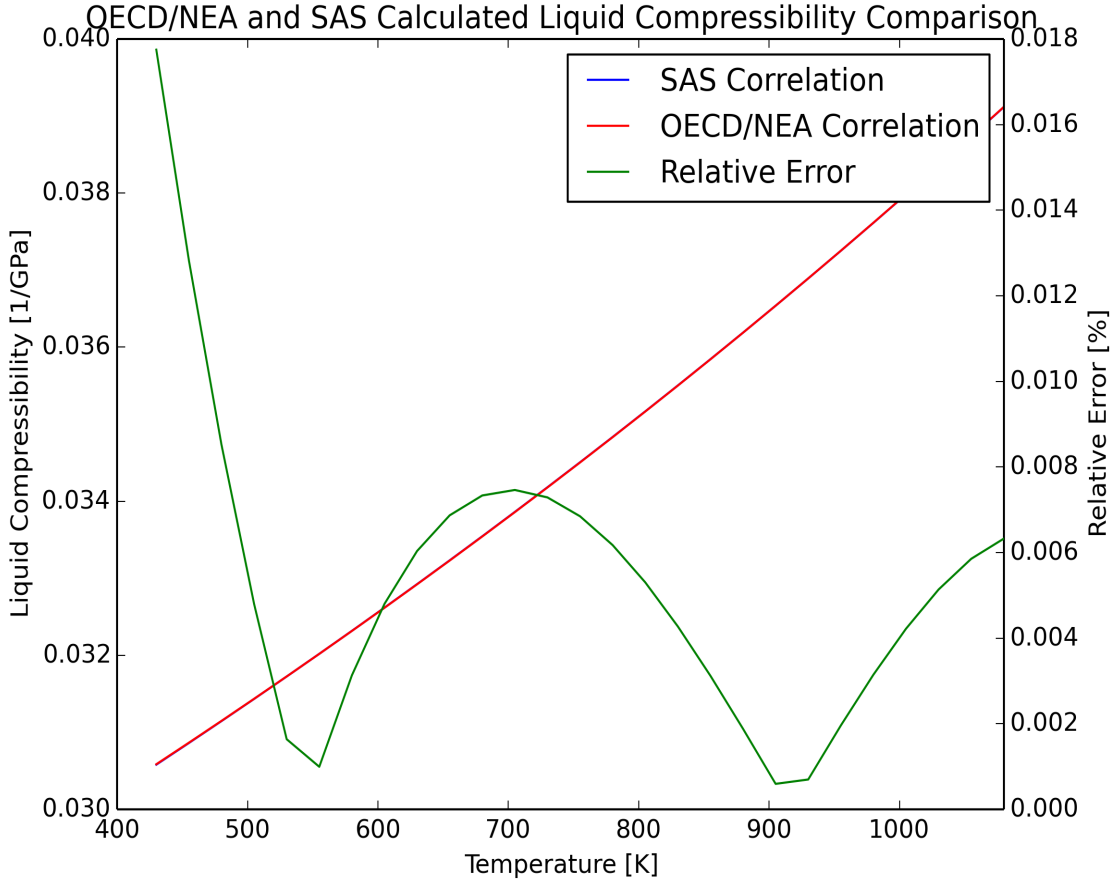
Reference *[A3.4-1]* indicates limited availability of sound velocity measurements in molten LBE, with tests conducted for temperatures [430–1100K]. Hence, applicability of the above correlation is considered to be [430–1100K], where the deviation from measured data is less than 1% in this temperature range as per *[A3.4-1]*.

The compressibility coefficient is calculated in SAS4A/SASSYS-1 by a polynomial expression based on the inverse of the coolant temperature subtracted from the critical temperature Eq. (3.19-61).

$$\beta_s = A_{40} + \frac{A_{41}}{(T_C - T)} \tag{3.19-93}$$

The mathematical expressions in *[A3.4-1]* and SAS4A/SASSYS-1 have different forms. The constant coefficients in the SAS4A/SASSYS-1 correlation were evaluated using a least squares equal weighted residual minimization fit to Eq. (3.19-92). The corresponding values for the coefficients of Eq. (3.19-93) are provided in Section 3.19.4.6.

Uncertainties for this correlation were not determined. However, absolute and relative errors with respect to Eq. (3.19-92) were calculated. The relative error is provided in the plot below.



3.19.4.4.10 LBE Thermal Expansion

The density of liquid metals changes with temperature due to thermal expansion related to anharmonicity of interatomic forces. As outlined in *[A3.4-1]*, the relationship between the density and the coefficient of volumetric thermal expansion (CVTE) at constant pressure p is defined by the following formula:

$$\alpha_p = -\frac{1}{\rho}\left(\frac{\partial \rho}{\partial T}\right)_p \tag{3.19-94}$$

Substituting the correlations for density recommended by *[A3.4-1]* yields:

$$\alpha_p = \frac{1}{8558 - T} \tag{3.19-95}$$

Since the thermal expansion correlation is defined from that of the density, the same temperature range of $[400 - 1273\text{K}]$ applies. The uncertainty of this quantity can be derived from the thermal expansion expression and the coolant density uncertainty.

The thermal expansion coefficient is calculated in SAS4A/SASSYS-1 by a polynomial expression based on the inverse of the coolant temperature subtracted to the critical temperature:

$$\alpha_p = A_{42} + \frac{A_{43}}{(T_C - T)} + \frac{A_{44}}{(T_C - T)^2} + \frac{A_{45}}{(T_C - T)^3} + \frac{A_{46}}{(T_C - T)^4} + \frac{A_{47}}{(T_C - T)^5} \tag{3.19-96}$$

A simple polynomial form matching between equations in Eq. (3.19-95) and Eq. (3.19-96) does not allow the SAS4A/SASSYS-1 expression to match that of *[A3.4-1]* with a reasonably small variation. Therefore, additional terms had to be included in the SAS4A/SASSYS-1 correlation. The correlation implemented includes terms from Eq. (3.19-96) up to the third order. The constant coefficients in the SAS4A/SASSYS-1 correlation were evaluated using a least squares equal weighted residual minimization fit to the OECD thermal expansion correlation. The corresponding values for the coefficients of Eq. (3.19-96) are provided in Section 3.19.4.6.

Uncertainties for this correlation were not determined. However, absolute and relative errors with respect to Eq. (3.19-95) were calculated. The relative error is provided in the plot below.

3.19.4.4.11 LBE Thermal Conductivity

Thermal conductivity is the fundamental thermophysical parameter of liquid metal coolants, which is used in all heat transfer calculations. The accuracy of its determination for liquid metals is poor and usually large discrepancies exist among different sets of data. Per *[A3.4-1]*, the following linear correlation was used as the thermal conductivity of the technically pure liquid LBE at normal pressure:

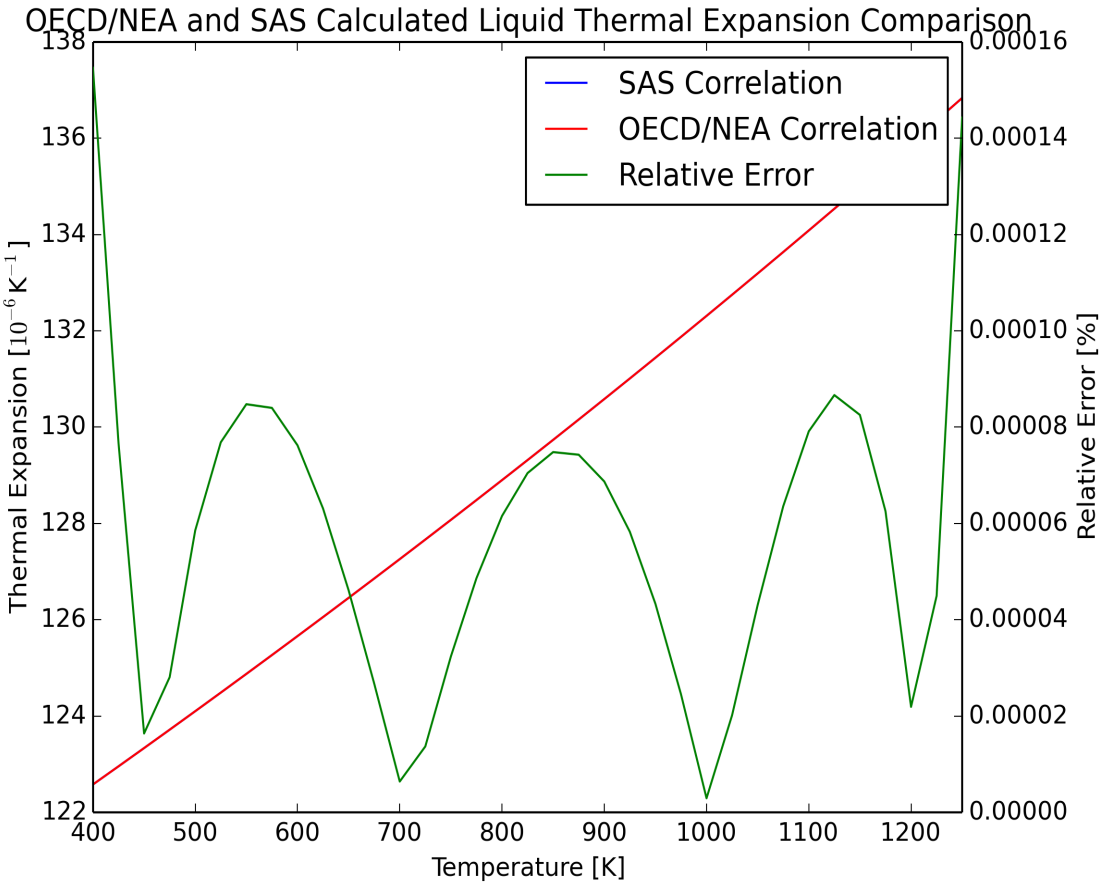
$$k_l = 3.284 + 1.617 \times 10^{-2} T - 2.305 \times 10^{-6} T^2 \tag{3.19-97}$$

The correlation is applicable over the range [400 – 1100 K] with a maximum deviation of 10-15%.

The SAS4A/SASSYS-1 mathematical expression correlating the thermal conductivity to the temperature is a third order polynomial as defined below:

$$k_l = A_{48} + A_{49} T + A_{50} T^2 + A_{51} T^3 \tag{3.19-98}$$

The corresponding values for the coefficients of Eq. (3.19-98) are provided in Section 3.19.4.6. The correlation accuracy and temperature ranges of Eq. (3.19-97) and Eq. (3.19-98) are consistent since the mathematical correlation form is maintained.


OECD/NEA and SAS Calculated Liquid Thermal Expansion Comparison
SAS Correlation
OECD/NEA Correlation
Relative Error
Thermal Expansion [10^{-6} K^{-1}]
Relative Error [%]
Temperature [K]
138
136
134
132
130
128
126
124
122
0.00016
0.00014
0.00012
0.00010
0.00008
0.00006
0.00004
0.00002
0.00000
400
500
600
700
800
900
1000
1100
1200

3.19.4.4.12 LBE Viscosity

Viscosity is a measure of friction among atoms of liquids. Because there is very little data on the impact of impurities on viscosity, the correlations below are for pure LBE. LBE is believed to be a Newtonian liquid. Therefore, the temperature dependence of their dynamic viscosity is usually described by an Arrhenius-type equation as described in *[A3.4-1]*.

$$\mu_l = 4.94 \times 10^{-4} e^{\left(\frac{754.1}{T}\right)} \tag{3.19-99}$$

This correlation is valid in the temperature range of $[400\text{–}1200\text{K}]$ and an estimated deviation of 6-8%.

Coolant dynamic viscosity is calculated in SAS4A/SASSYS-1 by a polynomial expression based on the inverse of the coolant temperature.

$$\mu_l = A_{52} + \frac{A_{53}}{T} + \frac{A_{54}}{T^2} + \frac{A_{55}}{T^3} \tag{3.19-100}$$

The mathematical expressions in Eq. (3.19-99) and Eq. (3.19-100) have different forms. The constant coefficients in the SAS4A/SASSYS-1 correlation were evaluated using a least squares equal weighted residual minimization fit to the OECD viscosity correlation. The corresponding values for the coefficients of Eq. (3.19-100) are provided in Section 3.19.4.6.

The temperature range over which this correlation was determined in *[A3.4-1]* is maintained because the least square fit to Eq. (3.19-99) was performed over the applicability range outlined above. Uncertainties for this correlation were not determined. However, absolute and relative errors with respect to Eq. (3.19-99) were calculated. The relative error is provided in the plot below.

3.19.4.5 Physical Properties of Other Coolants

The mathematical expressions utilized for the physical properties of sodium-potassium and heavy water coolants are identical to those described in *Lead Properties* and *Sodium Properties*. The corresponding coefficients utilized for all available built-in coolant properties are provided in Section 3.19.4.6. Additional details on formulation of these coefficients is not provided here.

3.19.4.6 SAS4A/SASSYS-1 Coolant Physical Properties

Table 3.19.24: Coolant Thermophysical Property Coefficients for SAS4A/SASSYS-1

Coefficient Name	Na	NaK	D_2O	Pb	Pb-Bi
Heat of Vaporization					
A1	5.3139E6	5.3139E6	3.5271E6	858600.0E0	8.560E5
A2	-2.0296E3	-2.0296E3	-7.3309E3	0.0E0	0.0E0
A3	1.0625E0	1.0625E0	15.229E0	0.0E0	0.0E0
A4	-3.3163E-4	-3.3163E-4	-.016126E0	0.0E0	0.0E0
Saturation Vapor Pressure					
A5	2.169E01	2.169E01	22.979E0	2.21678E+01	2.32247E1
A6	1.14846E4	1.14846E4	3504.1E0	2.11935E+04	2.25520E4

continues on next page

Table 3.19.24 – continued from previous page

Coefficient Name	Na	NaK	D_2O	Pb	Pb-Bi
A7	3.41769E5	3.41769E5	2.9432E5	5.35186E+05	7.04644E2
Saturation Temperature					
A8	[16]				
A9	[17]				
A10	[18]				
A11	[19]				
Liquid Density					
A12	1.00423E3	946.9E0	9.1568E2	11.441E+03	11.065E+03
A13	-0.2139E0	-0.2393E0	1.5447E0	-1.2795E0	-1.293E0
A14	-1.1046E-5	0.0E0	-3.0790E-3	0.0E0	0.0E0
Vapor Density					
A15	4.1444E-3	4.1444E-3	7.0429E-3	1.0E-02	1.0E-02
A16	-7.4461E-6	-7.4461E-6	-6.4776E-5	0.0E0	0.0E0
A17	1.3768E-8	1.3768E-8	3.5815E-7	0.0E0	0.0E0
A18	-1.0834E-11	-1.0834E-11	-9.7688E-10	0.0E0	0.0E0
A19	3.8903E-15	3.8903E-15	1.3034E-12	0.0E0	0.0E0
A20	-4.922E-19	-4.922E-19	-6.6481E-16	0.0E0	0.0E0
Saturation Pressure as function of Density					
A21	1.02801E3	1.02801E3	0.0E0	0.0E0	0.0E0
A22	4.3248E3	4.3248E3	0.0E0	0.0E0	0.0E0
A23	-1.5010E2	-1.5010E2	0.0E0	0.0E0	0.0E0
A24	2.8996E1	2.8996E1	0.0E0	0.0E0	0.0E0
A25	-2.9061E0	-2.9061E0	0.0E0	0.0E0	0.0E0
A26	1.4332E-1	1.4332E-1	0.0E0	0.0E0	0.0E0
A27	-2.7431E-3	-2.7431E-3	0.0E0	0.0E0	0.0E0
Liquid Heat Capacity					
A28	7.3898E5	0.0E0	0.0E0	1.0E0	1.04319E0
A29	3.1514E5	0.0E0	0.0E0	0.0E0	0.0E0
A30	1.1340E3	1834.0E0	5.1937E3	2.45422E+02	2.44470E2
A31	-2.2153E-1	-1.143E0	-7.1933E0	-6.67240E-02	-6.90385E-2
A32	1.1156E-4	3.391E-4	1.2557E-2	1.01474E-05	1.075363E-05
Vapor Heat Capacity					
A33	2.1409E3	2.1409E3	3.4488E4	0.0E0	0.0E0
A34	-2.2401E1	-2.2401E1	-2.8580E2	0.0E0	0.0E0
A35	7.9787E-2	7.9787E-2	9.4491E-1	0.0E0	0.0E0
A36	-1.0618E-4	-1.0618E-4	-1.1398E-3	0.0E0	0.0E0
A37	6.7874E-8	6.7874E-8	7.8111E-7	0.0E0	0.0E0
A38	-2.1127E-11	-2.1127E-11	0.0E0	0.0E0	0.0E0
A39	2.5834E-15	2.5834E-15	0.0E0	0.0E0	0.0E0
Liquid Adiabatic Compressibility					

continues on next page

Table 3.19.24 – continued from previous page

Coefficient Name	Na	NaK	D_2O	Pb	Pb-Bi
A40	-5.4415E-11	-5.4415E-11	4.4329E-12	-2.22038E-11	-1.82656E-11
A41	0.47663E-6	0.47663E-6	3.9575E-11	2.24037E-07	2.13456E-7
Liquid Thermal Expansion Coefficient					
A42	2.5156E-6	2.5156E-6	-4.8485E-2	1.79185E-05	1.50349E-5
A43	0.79919E0	0.79919E0	5.0206E1	6.63012E-01	7.05629E-1
A44	-6.9716E2	-6.9716E2	-1.9478E4	-1.13530E+03	-1.24294E3
A45	3.3140E5	3.3140E5	3.3867E6	8.44635E+05	9.69014E5
A46	-7.0502E7	-7.0502E7	-2.2082E8	1.0E0	1.0E0
A47	5.4920E9	5.4920E9	0.0E0	1.0E0	1.0E0
Liquid Thermal Conductivity					
A48	1.1045E2	14.18E0	-2.8088E-1	9.2E0	3.284E0
A49	-6.5112E-2	.03272E0	5.0999E-3	0.011E0	1.617E-2
A50	1.5430E-5	-2.202E-5	-8.0711E-6	0.0E0	-2.305E-6
A51	-2.4617E-9	0.0E0	2.5964E-9	0.0E0	0.0E0
Liquid Viscosity					
A52	3.6522E-5	-0.17049E-4	-5.8232E-3	3.48056E-04	3.86922E-4
A53	1.6626E-1	0.13434E0	7.9819161E0	8.67601E-01	6.54251E-1
A54	-4.56877E1	24.114E0	-3585.3469E0	-2.16495E+02	-1.15328E2
A55	2.8733E4	0.0E0	5.4455172E5	3.25911E+05	1.24838E5
Critical Temperature					
TCRIT	2.5033E3	2503.0E0	644.5E0	5000.0E0	4800.0E0

3.19.4.7 References

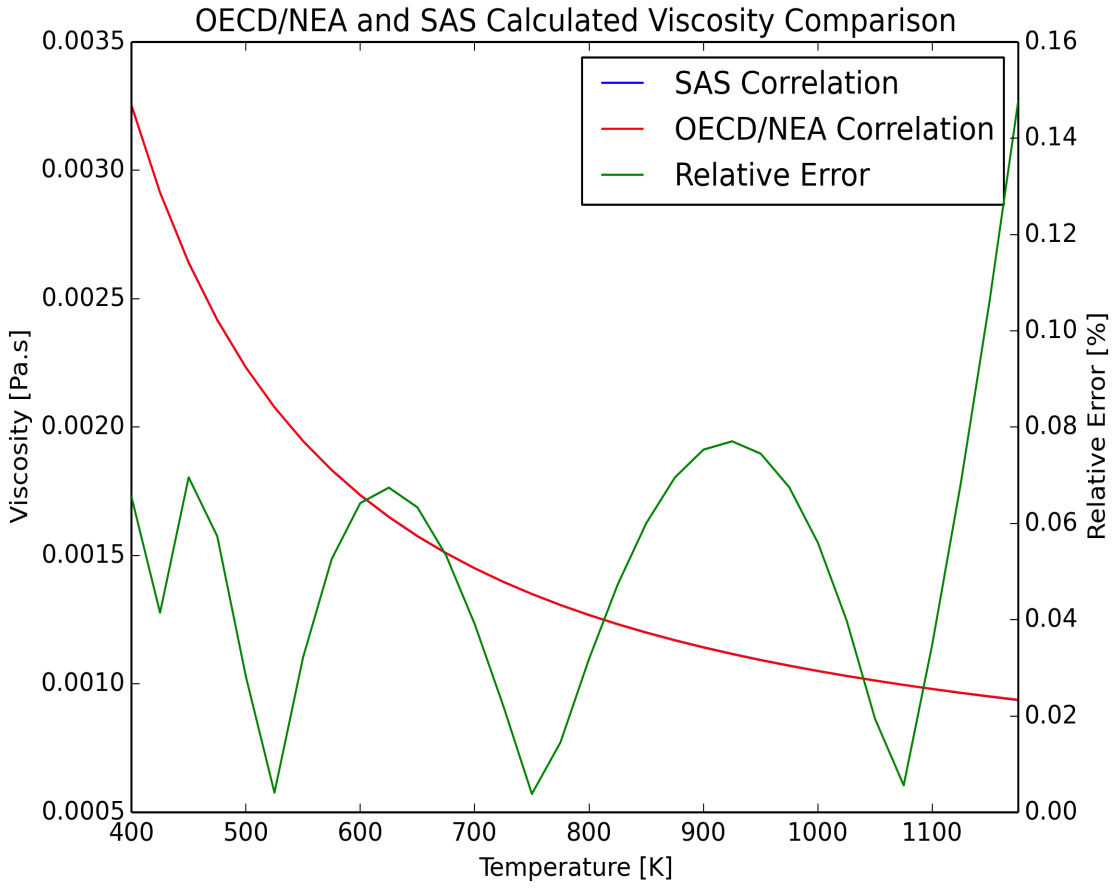
1. "Handbook on Lead-bismuth Eutectic Alloy and Lead Properties, Materials Compatibility, Thermal-hydraulics and Technologies", Nuclear Science, OECD/NEA, 2015.

[16] Coefficient A8, used in *[A3.4-2.4]*, is independent of the coolant type. It is defined as a function of the saturation pressure coefficients as: $A8 = 2 \times A7$.

[17] Coefficient A9, used in *[A3.4-2.4]*, is independent of the coolant type. It is defined as a function of the saturation pressure coefficients as: $A9 = -A6$.

[18] Coefficient A10, used in *[A3.4-2.4]*, is independent of the coolant type. It is defined as a function of the saturation pressure coefficients as: $A10 = A6^2 + 4 \times A5 \times A7$.

[19] Coefficient A11, used in *[A3.4-2.4]*, is independent of the coolant type. It is defined as a function of the saturation pressure coefficients as: $A11 = -4 \times A7$.


OECD/NEA and SAS Calculated Viscosity Comparison
SAS Correlation
OECD/NEA Correlation
Relative Error
Viscosity [Pa.s]
Relative Error [%]
Temperature [K]
0.0035
0.0030
0.0025
0.0020
0.0015
0.0010
0.0005
0.16
0.14
0.12
0.10
0.08
0.06
0.04
0.02
0.00
400
500
600
700
800
900
1000
1100

CHAPTER

FOUR

REACTOR POINT KINETICS, DECAY HEAT, AND REACTIVITY FEEDBACK

4.1 Introduction

The purpose of the SAS4A/SASSYS-1 reactor point kinetics, decay heat, and reactivity feedback models is to provide an estimate of the reactor power level to be used in the prediction of energy deposition in the fuel. Reactor material temperature changes and relocations determine the reactivity, which in turn determines the reactor power level and the rate of heating of the reactor materials.

The SAS4A/SASSYS-1 reactor point kinetics, and reactivity feedback models are based on concepts used in the SAS3A [4-1] computer code. A time-independent reactor power spatial shape is assumed, along with a space-independent (point) reactor kinetics model. However, the decay heat model in SAS4A/SASSYS-1 has been rewritten for version 5.0. First-order perturbation theory is used to predict reactivity feedback effects associated with material density changes. Fuel temperature (Doppler) effects are calculated assuming a logarithmic dependence on the local absolute temperature ratio, with a linearly dependent variation of the local Doppler coefficient on the coolant void fraction.

The fundamental basis for the assumptions of a time-independent power distribution, point kinetics, and first-order perturbation theory is the underlying supposition that the reactor neutron flux distribution is invariant in time. This means that in the transient simulation, the effects of changes in the reactor environment (geometry, dimensions, temperature and density distributions) on the neutron flux shape are neglected. This significantly reduces the complexity and computational expense of the overall neutronics model, with some loss of accuracy. In general, this inaccuracy can be expected to be significant mainly in the estimate of the reactivity feedback accompanying large-scale fuel material relocations, and large-scale fuel relocation is usually not included in a SAS4A/SASSYS-1 case.

The sections that follow describe the mathematical formulations for the total reactor power, delayed-neutron precursors, decay heat, and net reactivity. The numerical solution methods are described in Section 4.6, and subsequent sections provide details on code organization, data flow, input, and output.

4.2 Reactor Power

At any time t, the local power production at position $\overrightarrow{r}$ is assumed to be given by the space-time separated function:

$$Q\left(\overrightarrow{r},t\right)=\psi_{\mathrm{t}}\left(t\right)S\left(\overrightarrow{r}\right) \tag{4.2-1}$$

where $\psi_{\mathrm{t}}\left(t\right)$ is the dimensionless, normalized power amplitude and $S\left(\overrightarrow{r}\right)$ is the steady-state reactor power in watts being produced in an axial node at location $\overrightarrow{r}$. In terms of input quantities, $S\left(\overrightarrow{r}\right)$ is given by the

product of **POW** and **PSHAPE**. Initially, the power amplitude has a value of unity and $S(\vec{r})$ is normalized to the total steady-state reactor power. Appendix 4.1 contains a description of the internal normalization of PSHAPE performed by SAS4A/SASSYS-1. The time-dependent power amplitude is assumed to be made up of the sum of two components:

$$\psi_{\mathrm{t}}(t) = \psi_{\mathrm{f}}(t) + \psi_{\mathrm{h}}(t) \tag{4.2-2}$$

where $\psi_{\mathrm{h}}(t)$ comes from the decay of fission and capture products. These two components have been separated to allow the simulation of both short- and long-term transients.

The direct fission component of the power amplitude is given by

$$\psi_{\mathrm{f}}(t) = \psi_{\mathrm{f}}(0)\,\phi(t) \tag{4.2-3}$$

where $\phi(t)$ is the dimensionless, normalized fission power amplitude given by the point reactor kinetics model:

$$\dot{\phi}(t) = \phi(t)\,\frac{\delta k(t) - \beta}{\Lambda} + \sum_{\mathrm{i}} \lambda_{\mathrm{i}} C_{\mathrm{i}}(t) \tag{4.2-4}$$

with the initial condition $\phi(0) = 1$.

In Eq. (4.2-4), $\delta k(t)$ is the net reactivity, β is the total effective delayed-neutron fraction, Λ is the effective prompt neutron generation time, and λ_{i} is the decay constant for the delayed-neutron precursor isotope whose normalized population is $C_{\mathrm{i}}(t)$. The physical interpretation of the terms in the point reactor kinetics equation is made by Henry [4-2] and also by Bell and Glasstone [4-3].

4.3 Delayed-Neutron Precursors

The net rate of change of the delayed-neutron precursor population is given by

$$\dot{C}_{\mathrm{i}}(t) = \frac{\beta_{\mathrm{i}}\phi(t)}{\Lambda} - \lambda_{\mathrm{i}} C_{\mathrm{i}}(t) \tag{4.3-1}$$

where β_{i} is the effective delayed-neutron fraction for precursor i, and the initial, normalized steady precursor population is given by

$$C_{\mathrm{i}}(0) = \frac{\beta_{\mathrm{i}}}{\lambda_{\mathrm{i}}\Lambda} \tag{4.3-2}$$

In terms of the individual precursor delayed-neutron fractions, the total effective delayed-neutron fraction is given as

$$\beta = \sum_{\mathrm{i}} \beta_{\mathrm{i}} \tag{4.3-3}$$

The number of delayed neutron precursors is entered in input variable **NDELAY**, the effective delayed neutron fractions are entered in input array **BETADN**, and the delayed neutron precursor decay constants are entered in input array **DECCON**. The prompt neutron lifetime is entered in input variable **GENTIM**.

4.4 Decay Heat

A more detailed decay heat model has been developed for SAS4A/SASSYS-1 version 5.0 that allows greater flexibility than the previous model and can more accurately represent decay heat during long term transients. The key features of the new model include extending the number of exponential terms that can be used to represent decay heat from six to 24, the inclusion of built-in decay heat parameters from the most recent ANS decay heat power standard, the ability to mix multiple user-supplied and/or built-in decay heat curves within a single region, and the inclusion of a pre-defined non-decay-heat region.

In the following sections, the ANS decay heat standard is described, along with how it is applied to the SAS4A/SASSYS-1 decay heat model. The implementation of the new model with multiple decay heat curves per region is also discussed in this section, while a detailed description of existing and new input related to the updated decay heat model is provided in Section 4.8.

4.4.1 The ANS Decay Heat Power Standard

The American Nuclear Society (ANS) has published a standard for decay heat power in light water reactors.[4-4] For over 25 years, an extension to the standard has been planned to address fast reactor fuel cycles. In the absence of any such extension, the standard for light-water reactors is used for incorporation into the new decay heat model.

In the standard, decay heat power is defined for each of four different fissionable isotopes in terms of 23-term exponential functions:

$$f(t) = \sum_{\mathrm{n}=1}^{23} \alpha_{\mathrm{n}} e^{-\lambda_{\mathrm{n}} t} \quad \text{(MeV/Fission-s)} \tag{4.4-1}$$

where α_{n} is the immediate contribution (in MeV/s) of exponential term n to the decay power resulting from one fission event, λ_{n} is the decay constant for term n, and t is time in seconds after the fission event. The standard currently defines decay heat power for thermal fission in U-235 and Pu-239, fast fission in U-238, and thermal fission in Pu-241.

The ANS standard defines a method for calculating the decay heat power, after shutdown, which results from a known reactor power history. Although not stated in the standard, the method given makes an implicit assumption that the ratio between total power and fission power is fixed when calculating the number of fissions for a given power level. This is not precisely correct, especially for fresh fuel at the start of irradiation. An examination of the parameters given for U-235 suggest that it takes over ten hours of steady-state operation to reach 90% of the equilibrium decay power, and roughly 3½ months to reach 98%. At a fixed total power, then, the fission power will depend on the current level of the decay heat power. The standard avoids the issue by defining decay heat power on a per-fission basis, leaving it up to the user of the standard to provide an appropriate value for the recoverable energy per fission.

In light of this discrepancy and in consideration of the fact that SAS4A/SASSYS-1 performs decay heat calculations based on fission power (not total power) the methods prescribed in the ANS standard are modified so that they can be adapted for use in SAS4A/SASSYS-1. Furthermore, the standard describes “adjustments” that can be made to the calculated decay heat power to account for neutron capture in fission products. Because these adjustments are specific to a thermal-spectrum reactor, they are not accounted for here. In addition, the standard describes a method for including contributions from U-239 and Np-239 decay heat power. Because a fast reactor generally has a significant quantity of actinides contributing to decay heat power, this adjustment is also not accounted for here. Instead, user-supplied decay heat curves can be combined with the ANS standard curves to accomplish the same task in a manner that is relevant to the problem being solved.

4.4.1.1 Energy per Fission

The total *recoverable* energy per fission, Q_{t}, in a fissionable isotope consists of "prompt" fission energy, Q_{f}, and the energy from complete decay of fission products, Q_{d}:

$$Q_{\mathrm{t}} = Q_{\mathrm{f}} + Q_{\mathrm{d}} \quad \text{(MeV)} \tag{4.4-2}$$

The energy from decay of fission products can be calculated by integrating Eq. (4.4-1) over all time:

$$Q_{\mathrm{d}} = \sum_{\mathrm{n}=1}^{23} \frac{\alpha_{\mathrm{n}}}{\lambda_{\mathrm{n}}} \quad \text{(MeV)} \tag{4.4-3}$$

If the total recoverable energy per fission is known (or input by the user) then the prompt energy per fission can be determined as $Q_{\mathrm{f}} = Q_{\mathrm{t}} - Q_{\mathrm{d}}$. A useful term, to be used later, is the ratio of the decay energy to the prompt energy,

$$\beta_{\mathrm{d}} = \frac{Q_{\mathrm{d}}}{Q_{\mathrm{f}}} = \sum_{\mathrm{n}=1}^{23} \beta_{\mathrm{n}} = \sum_{\mathrm{n}=1}^{23} \frac{\alpha_{\mathrm{n}}}{\lambda_{\mathrm{n}} Q_{\mathrm{f}}} \tag{4.4-4}$$

Therefore, β_{d} is the total decay heat yield per unit fission heat. Note that for user-defined decay heat parameters, values for β_{n} and λ_{n} are defined by user input.

4.4.1.2 Number of Fissions

Eq. (4.4-1) gives the decay heat power as a function of time after a single fission event. The fission rate at time t' can be written as

$$F\left(t'\right) = P_0 \frac{P_{\mathrm{f}}\left(t'\right)}{Q_{\mathrm{f}} K} \quad \text{(Fissions/s)} \tag{4.4-5}$$

where P_0 is the nominal reactor power (Watts), f is the relative fission power normalized to the nominal reactor power, and $K = 1.602177 \times 10^{-13}$ J/MeV. The total number of fission events between time t' and $t' + \mathrm{dt}$ is then

$$F\left(t'\right) \mathrm{dt} = P_0 \frac{P_{\mathrm{f}}\left(t'\right)}{Q_{\mathrm{f}} K} \mathrm{dt}' \quad \text{(Fissions)} \tag{4.4-6}$$

4.4.1.3 Decay Power

The relative decay power at time $t > t'$ resulting from the fissions represented by Eq. (4.4-1) can be calculated by combining Eq. (4.4-6) with Eq. (4.4-1):

$$P_{\mathrm{d}}\left(t\right) = \frac{P_{\mathrm{f}}\left(t'\right)}{Q_{\mathrm{f}}} f\left(t - t'\right) \mathrm{dt}' \quad \left(t > t'\right) \tag{4.4-7}$$

where P_{d} is the relative decay power normalized to the nominal reactor power, P_0. The total decay power at time t from all fissions prior to time t can be determined by integrating the above equation:

$$P_{\mathrm{d}}\left(t\right) = \int_{-\infty}^{t} \frac{P_{\mathrm{f}}\left(t'\right)}{Q_{\mathrm{f}}} f\left(t - t'\right) \mathrm{dt}' \tag{4.4-8}$$

Note that Eq. (4.4-8) is similar in form to Equation (3) given in the ANS standard, with the exception that the ratio of total power to total recoverable energy, $P_{\mathrm{t}}/Q_{\mathrm{t}}$, is used in the ANS standard. The equation in the standard inaccurately calculates the fission rate in fresh fuel shortly after startup. While this is likely to be a very small error in most calculations, SAS4A/SASSYS-1 evaluates decay heat in terms of the fission power and the correct form of Eq. (4.4-8) is more convenient.

4.4.1.4 Contributions from Multiple Isotopes

In the preceding sections, a single fissionable isotope is implied. The determination of decay heat power contributions from multiple fissionable isotopes is straightforward. If x_{i} represents the fraction of the total fission power from isotope i, then the normalized fission power from isotope i is written as

$$P_{\mathrm{fi}} = x_{\mathrm{i}} P_{\mathrm{f}} \tag{4.4-9}$$

and the decay power from isotope i is

$$P_{\mathrm{di}}(t) = x_{\mathrm{i}} \int_{-\infty}^{t} \frac{P_{\mathrm{f}}(t')}{Q_{\mathrm{f}}} f(t-t')\,\mathrm{dt}' \tag{4.4-10}$$

Here, $f_{\mathrm{i}}(t)$ represents the decay heat parameters for isotope i.

In practice (as described in the next section on "Implementation") the decay heat power from isotope i is calculated using the normalized fission power only, as shown in Eq. (4.4-8), such that

$$P'_{\mathrm{di}}(t) = \int_{-\infty}^{t} \frac{P_{\mathrm{f}}(t')}{Q_{\mathrm{f}}} f(t-t')\,\mathrm{dt}' \tag{4.4-11}$$

The fractional fission powers are only applied when total (or integrated) decay heat power is calculated. Therefore, total decay heat power is defined as

$$P_{\mathrm{d}}(t) = \sum_{\mathrm{i}} x_{\mathrm{i}} P'_{\mathrm{di}}(t) \tag{4.4-12}$$

4.4.2 Implementation

Prior to version 5.0, a channel is assigned to a single decay heat curve by the parameter IDKCRV. Since multiple curves may now be assigned to a given channel, the parameter IDKCRV now refers to which decay heat *region* a channel belongs. The relationship between decay heat curves and decay heat regions is defined by new user input in the form of two input matrices, DKFRAC and DKANSI, that contains the various x_{i} for each region. With this, multiple curves can be assigned (with different weights) to a single region, and a single curve can be assigned to multiple regions. To support older input decks, the matrix defaults to an identity matrix where curve 1 is assigned to region 1, etc. In this case, curves and regions have the same meaning. A description of new and existing input is given in Section 4.8. In the following subsections, only a single decay heat curve is considered. Using the x_{i} provided in input, total (normalized) decay heat for each region can be calculated as described by Eq. (4.4-12).

4.4.2.1 Transient Calculations

At the beginning of a transient calculation, the initial decay heat power and fission power are known as a result of the steady-state initialization (see below). From that point on, fission power is determined based on the point (or spatial) kinetics equations. Given the normalized fission power, the normalized decay power can be solved using Eq. (4.4-8). To simplify notation, only a single term from Eq. (4.4-1) is considered, and a term for decay heat (i.e. energy) is introduced, where the decay heat, H_{n}, is defined by $P_{\mathrm{d}}(t) = \Sigma\ \lambda_{\mathrm{n}}\ H_{\mathrm{n}}(t)$. At time step k, the time varies from t_{k} to $t_{\mathrm{k}} + \tau$. Term n of Eq. (4.4-8) (in terms of decay heat) can then be written as

$$H_{\mathrm{n}}(t_{\mathrm{k}}+\tau) = e^{-\lambda_{\mathrm{n}}\tau} \int_{-\infty}^{t_{\mathrm{k}}} P_{\mathrm{f}}(t')\,\beta_{\mathrm{n}} e^{-\lambda_{\mathrm{n}}(t_{\mathrm{k}}-t')}\mathrm{dt}' + \int_{t_{\mathrm{k}}}^{t_{\mathrm{k}}+\tau} P_{\mathrm{f}}(t')\,\beta_{\mathrm{n}} e^{-\lambda_{\mathrm{n}}(t_{\mathrm{k}}+\tau-t')}\mathrm{dt}' \tag{4.4-13}$$

Note that the first integral is just the (known) decay heat at the beginning of the time step, $H_{\mathrm{n}}(t_{\mathrm{k}})$. To write the second integral in terms of the time within a time step, let $t = t' - t_{\mathrm{k}}$. Then the above equation can be written as

$$H_{\mathrm{n}}(t_{\mathrm{k}}+\tau) = H_{\mathrm{n}}(t_{\mathrm{k}})\, e^{-\lambda_{\mathrm{n}}\tau} + \beta_{\mathrm{n}} e^{-\lambda_{\mathrm{n}}\tau} \int_0^{\tau} P_{\mathrm{f}}^{k}(t_{\mathrm{k}}+t)\, e^{\lambda_{\mathrm{n}} t}\mathrm{dt} \tag{4.4-14}$$

From the point kinetics solution, the fission power over time step k is represented by a second-order polynomial:

$$P_{\mathrm{f}}^{k}(t_{\mathrm{k}}+t) = P_0 + P_1 t + P_2 t^2 \tag{4.4-15}$$

where $P_0 = P_{\mathrm{f}}(t_{\mathrm{k}})$. Using this expansion, the decay heat at the end of the point-kinetics time step can be solved:

$$H_{\mathrm{n}}(t_{\mathrm{k}}+\tau) = H_{\mathrm{n}}(t_{\mathrm{k}})\, e^{-\lambda_{\mathrm{n}}\tau} + \frac{\beta_{\mathrm{n}}}{\lambda_{\mathrm{n}}}\left[P_0 I_0(\tau) + \frac{P_1 I_1(\tau)}{\lambda_{\mathrm{n}}} + \frac{P_2 I_2(\tau)}{\lambda_{\mathrm{n}}^2}\right] \tag{4.4-16}$$

where

$$\begin{aligned} I_0(\tau) &= 1 - e^{-\lambda_{\mathrm{n}}\tau} \\ I_1(\tau) &= \lambda_{\mathrm{n}}\tau - I_0(\tau) \\ I_2(\tau) &= (\lambda_{\mathrm{n}}\tau)^2 - 2I_1(\tau) \end{aligned} \tag{4.4-17}$$

4.4.2.2 Steady-State Initialization

Prior to commencing with the transient calculations, SAS4A/SASSYS-1 performs a steady-state initialization that includes the determination of initial decay heat. The user has the option of entering a reactor power history as a histogram of relative reactor power. Assuming no initial decay heat is present (i.e. fresh fuel with no fission products) the initial fission power will be equal to the power level of the first histogram. However, as decay heat builds, fission power will decrease to maintain constant power.

Like the decay heat calculations during a transient, the decay heat during the constant-power interval, k, of duration τ_{k} can be written as

$$H_{\mathrm{n}}(t_{\mathrm{k}}+\tau_{\mathrm{k}}) = H_{\mathrm{n}}(t_{\mathrm{k}})\, e^{-\lambda_{\mathrm{n}}\tau_{\mathrm{k}}} + \beta_{\mathrm{n}} e^{-\lambda_{\mathrm{n}}\tau_{\mathrm{k}}} \int_0^{\tau_{\mathrm{k}}} P_{\mathrm{f}}^{\mathrm{k}}(t_{\mathrm{k}}+t)\, e^{\lambda_{\mathrm{n}} t}\mathrm{dt} \tag{4.4-18}$$

However, only the total power is known over the interval, not the fission power. The above equation can be rewritten as

$$H_{\mathrm{n}}(t_{\mathrm{k}}+\tau_{\mathrm{k}}) = H_{\mathrm{n}}(t_{\mathrm{k}})\, e^{-\lambda_{\mathrm{n}}\tau_{\mathrm{k}}} + \beta_{\mathrm{n}} e^{-\lambda_{\mathrm{n}}\tau_{\mathrm{k}}} \int_0^{\tau_{\mathrm{k}}} \left[P_{\mathrm{t}}^{k} - \sum_{\mathrm{n}=1}^{\mathrm{NDKGRP}} \lambda_{\mathrm{n}} H_{\mathrm{n}}(t_{\mathrm{k}}+t)\right] e^{\lambda_{\mathrm{n}} t}\mathrm{dt} \tag{4.4-19}$$

For a single fissionable isotope, this represents a coupled set of NDKGRP integral equations. When multiple fissionable isotopes contribute to a single decay heat region in the calculation, the number of coupled equations can increase dramatically.

To decouple the terms in Eq. (4.4-19), two assumptions are made. First, the fraction of fission power contributed by each fissionable isotope is assumed to be fixed by the values of x_{i}. Second, the fission power is assumed to be a constant such that the total power at the *end* of the constant-power interval matches the value supplied by the user. The first assumption only applies to decay heat regions that use more than one decay heat curve and may not be valid where there is significant depletion or breeding of fissionable isotopes during the course of steady-state initialization.

The second assumption was chosen for a number of reasons. First, by matching power at the end of the initiating power interval, it maintains continuity between the steady-state initialization and the transient calculation.

Second, it can be shown (by example) to introduce at most only a few hundredths of a percent error in most cases, and up to a few tenths of a percent error in very unusual situations. Third, it correctly predicts the two bounding cases of fresh fuel (no decay heat) and infinitely-long, steady-state equilibrium.

By assuming a constant (although initially unknown) fission power, the integral in Eq. (4.4-19) can be solved as before:

$$H_{\mathrm{n}}\left(t_{\mathrm{k}}+\tau\right)=H_{\mathrm{n}}\left(t_{\mathrm{k}}\right)e^{-\lambda_{\mathrm{n}}\tau}+P_{\mathrm{f}}\frac{\beta_{\mathrm{n}}}{\lambda_{\mathrm{n}}}\left(1-e^{-\lambda_{\mathrm{n}}t}\right) \tag{4.4-20}$$

Note that this is the same as Eq. (4.4-16) with $P_0 = P_{\mathrm{f}}$ and P_1 and P_2 set to zero. By multiplying by λ_{n} and summing the above equation over all decay terms, the decay *power* at the end of the time step is

$$P_{\mathrm{d}}\left(t_{\mathrm{k}}+\tau\right)=\sum_{\mathrm{n}}\lambda_{\mathrm{n}}H_{\mathrm{n}}\left(t_{\mathrm{k}}\right)e^{-\lambda_{\mathrm{n}}\tau}+P_{\mathrm{f}}\sum_{\mathrm{n}}\beta_{\mathrm{n}}\left(1-e^{-\lambda_{\mathrm{n}}t}\right) \tag{4.4-21}$$

For simplicity, the additional sum over isotopes with the appropriate x_{i}, as shown in Eq. (4.4-12), is implied.

Given the user-supplied total power and that $P_{\mathrm{t}} = P_{\mathrm{f}} + P_{\mathrm{d}}$, the fission power can now be solved:

$$P_{\mathrm{f}}=\frac{P_{\mathrm{t}}-\sum_{\mathrm{n}}\lambda_{\mathrm{n}}H_{\mathrm{n}}\left(t_{\mathrm{k}}\right)e^{-\lambda_{\mathrm{n}}\tau}}{1+\sum_{\mathrm{n}}\beta_{\mathrm{n}}\left(1-e^{-\lambda_{\mathrm{n}}t}\right)} \tag{4.4-22}$$

Again, the sum over isotopes is implied. If the preexisting decay power at the end of the time step (the second term in the numerator) is greater than the user-supplied total power, then the fission power for that histogram is set to zero by the code. This will be the case, for example, if the user specifies a zero-power shutdown period. Otherwise, the numerator represents the fission power if no new decay power is generated during the time step. The denominator then adjusts that value so that the fission power plus accumulated decay power will be equal to the total power at the end of the time step.

Finally, by substituting Eq. (4.4-22) into Eq. (4.4-20), the decay heat can be determined at the end of this step in the histogram. For an infinitely-long power interval, Eq. (4.4-22) simplifies to

$$P_{\mathrm{f}}=P_{\mathrm{t}}\frac{1}{1+\beta_{\mathrm{d}}} \tag{4.4-23}$$

and the decay heat from Eq. (4.4-20) is

$$H_{\mathrm{n}}\left(t_{\mathrm{k}}+\tau\right)=P_{\mathrm{f}}\frac{\beta_{\mathrm{n}}}{\lambda_{\mathrm{n}}} \tag{4.4-24}$$

These two expressions match the analytical solution exactly.

4.5 Net Reactivity

For applications other than EBR-II, the net reactivity in Eq. (4.2-4) is the sum of nine reactivity components:

$$\delta k\left(t\right)=\delta k_{\mathrm{p}}\left(t\right)+\delta k_{\mathrm{cs}}\left(t\right)+\delta k_{\mathrm{D}}\left(t\right)+\delta k_{\mathrm{d}}\left(t\right)+\delta k_{\mathrm{Na}}\left(t\right)+\delta k_{\mathrm{re}}\left(t\right)+\delta k_{\mathrm{cr}}\left(t\right)+\delta k_{\mathrm{fu}}\left(t\right)+\delta k_{\mathrm{cl}}\left(t\right) \tag{4.5-1}$$

where

δk_{p} = User-programmed reactivity,

δk_{cs} = Control system reactivity,

δk_{D} = Fuel Doppler feedback reactivity,

δk_{d} = Fuel, cladding, and structure axial expansion feedback reactivity,

δk_{Na} = Coolant density or voiding feedback reactivity,

δk_{re} = Core radial expansion feedback reactivity,

δk_{cr} = Control rod drive expansion feedback reactivity,

δk_{fu} = Fuel relocation reactivity feedback,

δk_{cl} = Cladding relocation reactivity feedback.

The reactivity feedback models available for these components are covered in the sections that follow. For EBR-II applications, the net reactivity is calculated as formulated in Section 4.5.9.

4.5.1 User-Programmed Reactivity

The user-programmed reactivity is specified by the user at execution time as either an input table or as subprogram FUNCTION PREA. It is intended to be used as a means of specifying any reactivity effect not explicitly modeled as a feedback. An example might be the simulation of a control rod withdrawal or insertion, or the dropping of a fuel subassembly during reloading.

The user-programmed reactivity option is triggered by setting **IPOWER** to 0. For **NPREAT** equal to 0, the value returned by the user-supplied subroutine function PREA is used as the programmed reactivity. For **NPREAT** > 0, **NPREAT** gives the number of pairs of values of programmed reactivity and time input on the standard input file in **PREATB** and **PREATM**. A maximum of twenty pairs of programmed reactivity and time may be entered in **PREATB** and **PREATM**. These data are then either used directly or fit to curves according to the user's specification of input data **IFIT**.

4.5.2 Control System Reactivity

The control system reactivity is the value supplied by the reactivity control signal (JTYPE = -1) generated by the control system model (see Chapter 6 and Input Block 5-INCONT).

4.5.3 Fuel Doppler Feedback Reactivity

The fuel Doppler reactivity effect at any axial location in a subassembly is estimated from

$$T\frac{\mathrm{d}\left(\delta k_{\mathrm{D}}\right)}{\mathrm{dT_f}} = \alpha_{\mathrm{D}} \tag{4.5-2}$$

where T_{f} is the radial mass-averaged fuel temperature for that axial location, and α_{D} is the local fuel Doppler coefficient, an input quantity. To obtain the Doppler reactivity feedback at time t, Eq. (4.5-2) is integrated from steady-state conditions to conditions at time t to obtain

$$\delta k_{\mathrm{D}}\left(t\right) = \alpha_{\mathrm{D}} \ln\left[\frac{T_{\mathrm{f}}\left(t\right)}{T_{\mathrm{f}}\left(0\right)}\right] \tag{4.5-3}$$

Eq. (4.5-3) is used at each axial location where a fuel temperature is calculated. The local fuel Doppler coefficient, α_{D}, is adjusted linearly between the input coolant-in (flooded) and coolant-out (voided) values to correct for the effect of coolant voiding on neutron leakage.

The coolant-in and coolant-out Doppler coefficients are entered as **ADOP** and **BDOP**, and the axial weighting of the Doppler coefficients is input as **WDOPA**.

4.5.4 Fuel, Cladding, and Structure Axial Expansion Feedback Reactivity

4.5.4.1 Simple Axial Expansion Reactivity Model

A simple model for the reactivity effects of thermal expansion of fuel and cladding is included in SAS4A/SASSYS-1. The simple thermal expansion feedback model is based on a few assumptions. It is assumed that before the start of the transient, a combination of fuel cracking, fuel re-structuring, and stress relaxation cause the gap between the fuel and the cladding to close, but there is little contact force between the fuel and the cladding. During the transient, if the cladding expands faster than the fuel, then the fuel-cladding gap opens, and the fuel can expand freely in the axial direction. If the fuel expands faster than the cladding, then the fuel binds with the cladding, and the axial expansion is determined by balancing the axial forces between the fuel and the cladding. Slip between fuel and cladding is ignored in this case. The axial expansion feedback is calculated separately for each channel. The formulation for each channel follows.

For each axial node, j, in the core, the fuel and cladding expansion fractions are calculated as

$$\epsilon_{\mathrm{f}}(j,t) = \alpha_{\mathrm{f}}\left[T_{\mathrm{f}}(j,t) - T_{\mathrm{f}}(j,0)\right] \tag{4.5-4}$$

and

$$\epsilon_{\mathrm{e}}(j,t) = \alpha_{\mathrm{e}}\left[T_{\mathrm{e}}(j,t) - T_{\mathrm{e}}(j,0)\right] \tag{4.5-5}$$

where

α_{f} = fuel thermal expansion coefficient

α_{e} = cladding thermal expansion coefficient

T_{f} = mass-averaged fuel temperature for axial node j

and

T_{e} = cladding mid-point temperature

If $\epsilon_{\mathrm{e}}(j)$ is greater than $\epsilon_{\mathrm{f}}(j)$, then the fuel axial expansion for node j is

$$\delta_{\mathrm{f}}(j,t) = \epsilon_{\mathrm{f}}(j,t)\, D_{\mathrm{z}}(j) \tag{4.5-6}$$

and the cladding expansion is

$$\delta_{\mathrm{e}}(j,t) = \epsilon_{\mathrm{e}}(j,t)\, D_{\mathrm{z}}(j) \tag{4.5-7}$$

where D_{z} is the nominal axial height of the node. If $\epsilon_{\mathrm{f}}(j)$ is greater than or equal to $\epsilon_{\mathrm{e}}(j)$, then a simple balance between the axial forces of the fuel and cladding gives

$$\delta_{\mathrm{f}}(j,t) = \frac{\epsilon_{\mathrm{f}}(j,t)\, Y_{\mathrm{f}} A_{\mathrm{f}} + \epsilon_{\mathrm{e}}(j,t)\, Y_{\mathrm{e}} A_{\mathrm{e}}}{Y_{\mathrm{f}} A_{\mathrm{f}} + Y_{\mathrm{e}} A_{\mathrm{e}}} D_{\mathrm{z}}(j) \tag{4.5-8}$$

and

$$\delta_{\mathrm{e}}(j,t) = \delta_{\mathrm{f}}(j,t) \tag{4.5-9}$$

where

Y_{f} = fuel Young's modulus

Y_{e} = cladding Young's modulus

A_{f} = nominal cross-sectional area of the fuel

and

A_e = nominal cross-sectional area of the cladding

The reactivity calculation is based on the fuel and cladding worth tables used by the code. First an unexpanded axial mesh, $z_0(j)$, is calculated using

$$z_0(1) = 0 \tag{4.5-10}$$

and

$$z_0(j+1) = z_0(j) + D_z(j) \tag{4.5-11}$$

Note that z_0 is zero at the bottom of the lower axial blanket.

New, expanded axial meshes for the fuel and cladding, $z_{nf}(j)$ and $z_{ne}(j)$, are calculated using

$$z_{nf}(1) = 0 \tag{4.5-12}$$

$$z_{ne}(1) = 0 \tag{4.5-13}$$

$$z_{nf}(j+1) = z_{nf}(j) + f_f(j) D_z(j) \tag{4.5-14}$$

and

$$z_{ne}(j+1) = z_{ne}(j) + f_e(j) D_z(j) \tag{4.5-15}$$

where

$$f_f(j) = 1 + \frac{\delta_f(j,t)}{D_z(j)} \tag{4.5-16}$$

and

$$f_e(j) = 1 + \frac{\delta_e(j,t)}{D_z(j)} \tag{4.5-17}$$

Users define axially dependent fuel worth per unit mass, $R_f(j)$, and axially dependent cladding worth per unit mass, $R_e(j)$, via the inputs **FUELRA** and **CLADRA**, respectively. Both are defined on the original mesh, $z_0(j)$.

Alternatively, the axially dependent fuel worth can be defined as a multi-component mixture using mass fractions and reactivity feedback coefficients for each component of the fuel. This option can be invoked by defining references to additional tables of mass fraction and reactivity feedback data using the inputs **iMassFractionTable** and **iReactivityTable**. Specifically, this allows for the modeling of axially heterogeneous fuel or other more complex scenarios. When this option is used, the overall fuel worth per unit mass for fuel with k components is

$$R_f(j) = \sum_k f_k(j) r_k(j) \tag{4.5-18}$$

where

$f_k(j)$ = mass fraction of fuel component k at axial node j,

$r_k(j)$ = fuel worth per unit mass of fuel component k at axial node j.

Currently, the multi-component fuel mass fractions and reactivity feedback tables are not compatible with the fuel relocation models.

If node j has been shifted and expanded so that

$$z_0(j) \leq z_{\text{nf}}(j) \leq z_0(j+1) \tag{4.5-19}$$

and

$$z_0(j+1) \leq z_{\text{nf}}(j+1) \leq z_0(j+2) \tag{4.5-20}$$

as in Figure 4.5.1, then $\Delta\rho_{\text{f}}(j)$, the fuel contribution to axial expansion feedback from node j is calculated as

$$\Delta\rho_{\text{f}}(j) = m_{\text{f}}(j) R_{\text{f}}(j) \frac{z_0(j+1) - z_{\text{nf}}(j)}{z_{\text{nf}}(j+1) - z_{\text{nf}}(j)} + m_{\text{f}}(j) R_{\text{f}}(j+1) \frac{z_{\text{nf}}(j+1) - z_0(j+1)}{z_{\text{nf}}(j+1) - z_{\text{nf}}(j)} - m_{\text{f}}(j) R_{\text{f}}(j) \tag{4.5-21}$$

which can also be written as

$$\Delta\rho_{\text{f}}(j) = R_{\text{f}}(j) \Delta m_{\text{f}}(j) \tag{4.5-22}$$

A similar expression is used to calculate $\Delta\rho_{\text{e}}(j)$, the cladding contribution to axial expansion feedback. If $z_{\text{nf}}(j+1)$ has expanded past $z_0(j+2)$, then a summation over the appropriate nodes is used instead of Eq. (4.5-21).

For the cladding, second-order axial expansion effects can optionally be considered by defining additional columns in the **iReactivityTable**. When using the second-order model, the cladding contribution to axial expansion feedback becomes

$$\Delta\rho_{\text{e}}(j) = R_{\text{e}}(j) \Delta m_{\text{e}}(j) + R_{\text{e,2}}(j) [\Delta m_{\text{e}}(j)]^2 \tag{4.5-23}$$

where $R_{\text{e,2}}(j)$ is the user-defined second-order coefficient (column *CLADRA2* of **iReactivityTable**). Currently, second-order cladding axial expansion effects cannot be used with fuel and cladding relocation models.

By default, SAS4A/SASSYS-1 calculates the axial expansion of fuel and cladding as described above. In addition to the gap-dependent model described above, the user can choose cladding-controlled expansion, independent free expansion, or force balance controlled expansion at all times. These options are selected by the input **MODEEX**.

Cladding-controlled expansion may be appropriate for cases where there is no gap between fuel and cladding and the fuel Young's modulus is much less than the cladding Young's modulus. In this case, the cladding expansion is calculated by Eq. (4.5-7), and the fuel expansion is set equal to the cladding expansion. Independent free expansion may be appropriate for cases where the gap is expected to be maintained throughout the transient; the expansion is then determined by Eq. (4.5-6) and Eq. (4.5-7), without the need for force balance. The continual force balance option may be appropriate for cases where the fuel and cladding are expected to be in contact at the initial condition, and remain in contact throughout the transient. In which case, expansion is determined by Eq. (4.5-8) and Eq. (4.5-9), regardless of the relative expansion rates of fuel and cladding.

A feature has been added to include the reactivity effect associated with the axial thermal expansion of structure during the transient. This feature may be used, for instance, to treat the axial thermal expansion of the subassembly duct walls.

Structure is considered to be free, and expansion is calculated in an analogous way as the independent free expansion model for fuel and cladding. That is, equations for the structure expansion are written similar to Eq. (4.5-5) and Eq. (4.5-7), with no need for force balance. An expression for the reactivity change in node j, $\Delta\rho_{\text{s}}(j)$, is written similar to Eq. (4.5-21), based on changes in the grid calculated in an analogous fashion as Eq. (4.5-15). Similar to cladding axial expansion, second-order axial structure expansion effects can optionally

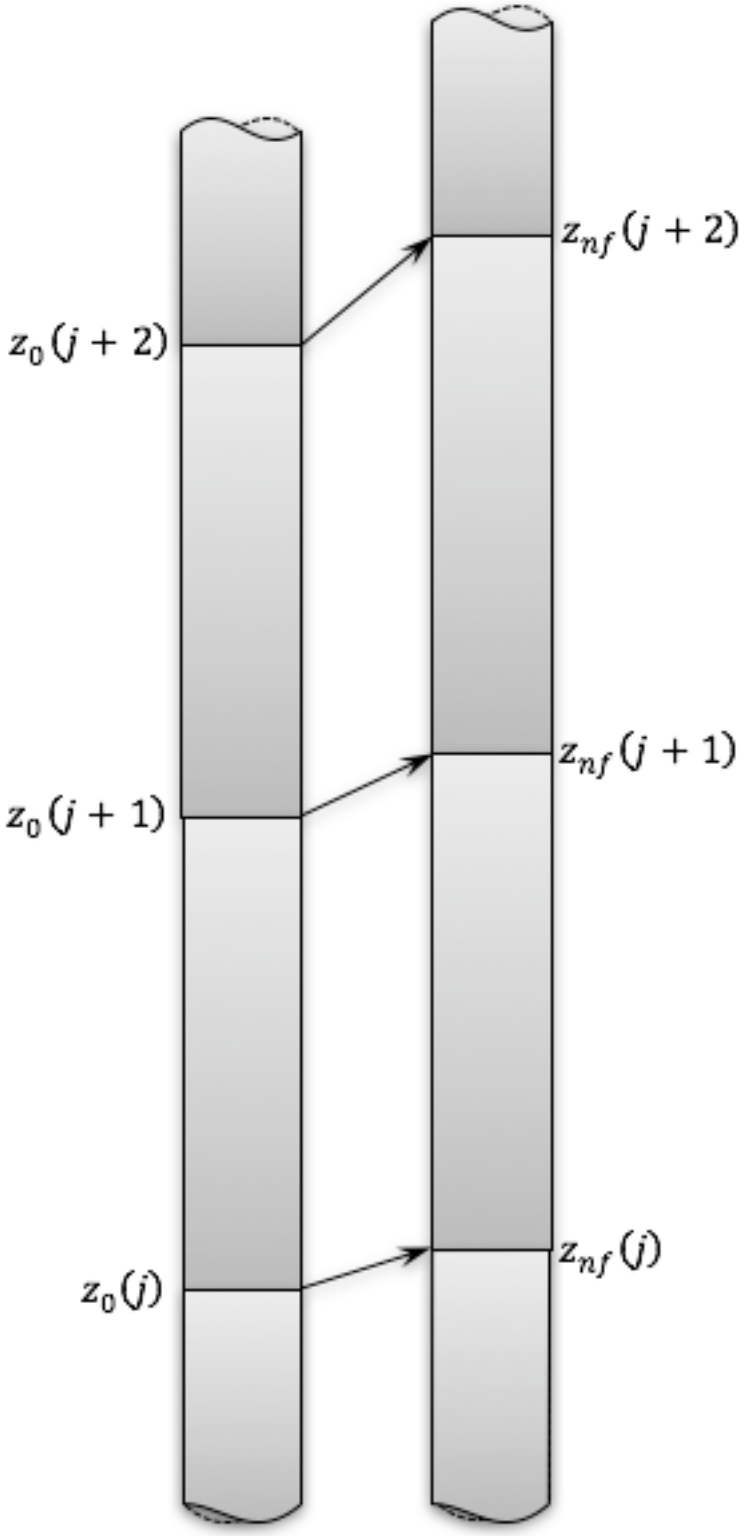


Figure 4.5.1: Original and Expanded Fuel Axial Meshes

be considered by defining additional columns in the **iReactivityTable**. In this case, the corresponding equation is Eq. (4.5-23).

The total reactivity change, δk_{d}, is

$$\delta k_{\mathrm{d}} = \epsilon_{\mathrm{ex}} \sum_{\mathrm{j}} \left[\Delta\rho_{\mathrm{f}}\left(j\right) + \Delta\rho_{\mathrm{e}}\left(j\right) + \Delta\rho_{\mathrm{s}}\left(j\right) \right] \tag{4.5-24}$$

where ϵ_{ex} is an effective axial expansion multiplier defined by **EXPCFF**.

The summation in Eq. (4.5-24) is only over the core nodes. The axial blankets are ignored in the fuel expansion feedback calculations. In order to obtain an accurate value for the axial expansion reactivity feedback, the fuel worth input for the upper axial blanket nodes must be the worth of core fuel in the blanket region.

The cladding, fuel, and structure worth used in Eq. (4.5-21) and the first-order cladding and structure worth used in Eq. (4.5-23) can be input in arrays **CLADRA**, **FUELRA**, and **STRCRA**, or using a TABLE block referenced by the **iReactivityTable** parameter. Second-order coefficients for cladding or structure axial expansion feedback used in Eq. (4.5-23) are input in the TABLE block referenced by the **iReactivityTable** parameter. The table block input format is described in Table 4.5.1. Mass fractions for multi-component fuel compositions can be defined in a TABLE block referenced by the **iMassFractionTable** parameter. The table block input format for a mass fraction table is described in Table 4.5.2. If **iMassFractionTable** is 0 then the fuel is assumed to be a single component.

Table 4.5.1: Input in a Reactivity Table

Column Label	Description	Units
LENGTH (optional)	Axial lengths of the mesh intervals over which the reactivity feedback data in this table applies. The total length of all mesh intervals must match the length of the MZ or MZC mesh, depending on the value of **IREACZ**. If the LENGTH column is omitted, then the mesh is assumed to be the same as the MZ or MZC mesh, according to **IREACZ**.	m
FUELRA (optional)	Fuel reactivity worth for each axial mesh interval. If this column is present, the values for **FUELRA** are ignored.	Δk/k-kg
CLADRA (optional)	Clad reactivity worth for each axial mesh interval. If this column is present, the values for **CLADRA** are ignored.	Δk/k-kg
CLADRA2 (dependent)	Second-order clad reactivity worth for each axial mesh interval. May only be present if *CLADRA* is present. Assumed to be 0.0 if not input.	Δk/k-kg^2
VOIDRA (optional)	*Void reactivity worth* for each axial mesh interval. If this column is present, the values for **VOIDRA** are ignored.	Δk/k-kg
VOIDRA2 (dependent)	*Second-order void reactivity worth* for each axial mesh interval. May only be present if *VOIDRA* is present. Assumed to be 0.0 if not input.	Δk/k-kg^2
STRCRA (optional)	Structure reactivity worth for each axial mesh interval. If this column is present, the values for **STRCRA** are ignored.	Δk/k-kg
STRCRA2 (dependent)	Second-order structure reactivity worth for each axial mesh interval. May only be present if *STRCRA* is present. Assumed to be 0.0 if not input.	Δk/k-kg^2
<any> (optional)	Reactivity worth distribution(s) for multi-component fuel compositions. The interpretation of additional columns in this table is determined by the definition of the *fuel mass fraction table*.	Δk/k-kg

Example reactivity table input:

```
TABLE <id> ReactivityTable
     FUELRA      CLADRA      VOIDRA      FUELRA2
     0.0          0.0         0.0         0.0        ! Lower Reflector
     1.1E-05      0.0         0.0         1.1E-05    ! Fuel, Segment 1
     1.3E-05      0.0         0.0         1.3E-05    ! Fuel, Segment 2
     1.6E-05      0.0         0.0         1.6E-05    ! Fuel, Segment 3
     1.9E-05      0.0         0.0         1.9E-05    ! Fuel, Segment 4
     2.3E-05      0.0         0.0         2.3E-05    ! Fuel, Segment 5
     2.4E-05      0.0         0.0         2.4E-05    ! Fuel, Segment 6
     2.7E-05      0.0         0.0         2.7E-05    ! Fuel, Segment 7
     3.1E-05      0.0         0.0         3.1E-05    ! Fuel, Segment 8
     3.6E-05      0.0         0.0         3.6E-05    ! Fuel, Segment 9
     4.0E-05      0.0         0.0         4.0E-05    ! Fuel, Segment 10
     3.7E-05      0.0         0.0         3.7E-05    ! Fuel, Segment 11
     3.3E-05      0.0         0.0         3.3E-05    ! Fuel, Segment 12
     2.5E-05      0.0         0.0         2.5E-05    ! Fuel, Segment 13
     1.9E-05      0.0         0.0         1.9E-05    ! Fuel, Segment 14
     7.0E-05      0.0         0.0         7.0E-05    ! Fuel, Segment 15
     1.4E-05      0.0         0.0         1.4E-05    ! Fuel, Segment 16
     1.0E-05      0.0         0.0         1.0E-05    ! Gas Plenum
     0.0          0.0         0.0         0.0        ! Upper Reflector
END
```

Table 4.5.2: Input in a Mass Fraction Table

Column Label	Description
<any> (optional)	Mass fraction of fuel component "any" for each axial mesh interval in the original fuel (MZ) mesh. Column labels in the mass fraction table must match a column label in the *reactivity table*.

Note that the following terms are considered reserved words and may not be used as column labels: LENGTH, CLADRA, CLADRA2, VOIDRA, VOIDRA2, STRCRA, STRCRA2. Any other label, including FUELRA, is permitted.

Example fuel mass fraction table input:

```
TABLE <id> MassFractionTable
     FUELRA      FUELRA2
      0.3         0.7     ! Lower Reflector
      0.3         0.7     ! Fuel, Segment 1
      0.3         0.7     ! Fuel, Segment 2
      0.3         0.7     ! Fuel, Segment 3
      0.3         0.7     ! Fuel, Segment 4
      0.3         0.7     ! Fuel, Segment 5
      0.3         0.7     ! Fuel, Segment 6
      0.3         0.7     ! Fuel, Segment 7
      0.3         0.7     ! Fuel, Segment 8
      0.3         0.7     ! Fuel, Segment 9
      0.3         0.7     ! Fuel, Segment 10
      0.3         0.7     ! Fuel, Segment 11
      0.3         0.7     ! Fuel, Segment 12
      0.3         0.7     ! Fuel, Segment 13
      0.3         0.7     ! Fuel, Segment 14
      0.3         0.7     ! Fuel, Segment 15
```

(continues on next page)

(continued from previous page)

```
      0.3         0.7    ! Fuel, Segment 16
      0.3         0.7    ! Gas Plenum
      0.3         0.7    ! Upper Reflector
END
```

The effective axial expansion multiplier in Eq. (4.5-24) is input as **EXPCFF**. Thermal expansion coefficients for fuel, cladding, and structure are input in **FUELEX**, **CLADEX**, and **StructEX**. The initial mass of cladding and structure in each channel is computed using the density **DENSS**. The simple axial expansion reactivity model is invoked by **IAXEXP** with the axial expansion mode option selection in **MODEEX**.

4.5.4.2 DEFORM-4 Axial Expansion Reactivity Model

For any channel in which the DEFORM-4 module (see Chapter 8) has been specified, the axial expansion reactivity feedback will be calculated as described in Eq. (4.5-10) through Eq. (4.5-24) but using fuel and cladding expanded axial mesh heights as calculated by DEFORM-4. The DEFORM-4 effective axial expansion multiplier is entered as **EXPCOF**.

4.5.5 Coolant Density Feedback Reactivity

Reactivity feedback effects from either single-phase coolant density changes or two-phase coolant boiling are calculated using the input coolant void reactivity worth table **VOIDRA** or a TABLE block referenced by **iReactivityTable** that includes a *VOIDRA* column (see Table 4.5.1). The reactivity feedback from coolant density and voiding changes is calculated from

$$\delta k_{\text{Na}} = \epsilon_{\text{d}} \sum_i \sum_j (\rho_c)_{ij} \, \alpha_{ij} \tag{4.5-25}$$

where

ϵ_{d} = sodium density multiplier

$(\rho_c)_{ij}$ = coolant void worth in axial segment j of channel i

and

α_{ij} = average coolant void fraction in segment j of channel i.

Second-order effects can optionally be considered, in which case the reactivity feedback from coolant density changes is calculated using

$$\delta k_{\text{Na}} = \epsilon_{\text{d}} \sum_i \sum_j \left[(\rho_c)_{ij} \, \alpha_{ij} + (\rho_{c,2})_{ij} \, \alpha_{ij}^2 \right] \tag{4.5-26}$$

where $\rho_{c,2}$ is the second order coefficient. Second-order coefficients are input using a TABLE block referenced by **iReactivityTable** that includes a *VOIDRA2* column (see Table 4.5.1). Currently, second-order void effects are not compatible with the sodium boiling model.

The local coolant void fraction is calculated based on either a) liquid coolant density changes from the initial steady-state condition, or b) combined liquid density and boiling-induced voiding density changes from the steady-state condition. The sodium density multiplier is entered as **COLFAC**.

4.5.6 Radial Expansion Feedback Reactivity

4.5.6.1 Simple Radial Expansion Reactivity Model

Two radial expansion feedback models are available in SAS4A/SASSYS-1: a simple model described here and a more detailed model described in Section 4.5.6.2.

The simple radial expansion feedback model in SAS4A/SASSYS-1 is based on a model by Huebotter [4-4]. The radial growth of the core is determined by the expansion of the lower grid support structure and by the expansion of the duct walls at the above core load pads. The expansion of the lower grid support structure is assumed to be proportional to the rise in the subassembly inlet temperature above its initial steady-state value. The expansion at the location of the above core load pads is assumed to be proportional to the change in the average structure temperature at this location.

The equations actually used in SAS4A/SASSYS-1 are

$$\delta k_{\text{re}} = C_{\text{re}} \left[\Delta T_{\text{in}} + \frac{\text{XMC}}{\text{XAC}} \left(\Delta \overline{T}_{\text{SLP}} - \Delta T_{\text{in}} \right) \right] \tag{4.5-27}$$

where

t = time, s

δk_{re} = reactivity change due to radial expansion, \$

C_{re} = radial expansion coefficient, \$/K

$\Delta T_{\text{in}} = T_{\text{in}}(t) - T_{\text{in}}(t_1)$, K

T_{in} = coolant inlet temperature, K

t_1 = time at the end of the first main time step, s

XMC = distance from nozzle support point to core midplane, m

XAC = distance from nozzle support point to above core load pad, m

$$\Delta \overline{T}_{\text{SLP}} = \overline{T}_{\text{SLP}}(t) - \overline{T}_{\text{SLP}}(t_1) \quad K \tag{4.5-28}$$

$$\overline{T}_{\text{SLP}}(t) = \frac{\sum_{\text{i}} N_{\text{s}}(i)\, T_{\text{SLP}}(i,t)\, f_{\text{i}}}{\sum_{\text{i}} N_{\text{s}}(i)\, f_{\text{i}}} \tag{4.5-29}$$

$\overline{T}_{\text{SLP}}$ = average structure temperature at the above core load pads

$T_{\text{SLP}}(i,t)$ = structure temperature (outer structural radial node) in channel *i* at the axial node corresponding to the above core load pad

i = channel number

$N_{\text{s}}(i)$ = number of subassemblies represented by channel i

$$f_{\text{i}} = \begin{cases} 1 & \text{if channel is to be included in the average} \\ 0 & \text{if channel is not to be included} \end{cases}$$

An option has been included in the code to use the inlet plenum wall temperature instead of the coolant inlet temperature for T_{in}. This option can be used to account for time delays in the heating of the lower grid support structure. Note that for coding simplicity the temperatures at the end of the first main time step are used as the reference values, rather than using steady-state temperatures for the reference. This makes very little difference in the results, since the changes in the inlet temperature and the outer structure node temperature above the core are normally extremely small during the first time step. Also, the user can choose which channels will be included in the averaging of Eq. (4.5-29) by specifying f_{i} for each channel using **IRDEXP**.

This model was not explicitly set up to account for subassembly bowing or flowering of the core, but the user can set arbitrary values for C_{re} and $\left[\frac{\text{XMC}}{\text{XAC}}\right]$ in Eq. (4.5-27). Therefore, if the bowing reactivity effect is proportional to $\Delta\overline{T}_{\text{SLP}}$ or to $\Delta\overline{T}_{\text{SLP}} - \Delta\overline{T}_{\text{in}}$, then bowing reactivity can be accounted for by adjusting C_{re} and $\frac{\text{XMC}}{\text{XAC}}$, which are entered as input variables **RDEXPC** and **XMCXAC**. The simple radial expansion model is invoked by specifying **IRADEX**.

4.5.6.2 Detailed Radial Expansion Reactivity Model

The basic radial core expansion reactivity feedback model described in Section 4.5.6.1 incorporated several major assumptions that restricted the ability of the model to calculate the reactivity feedback accurately, particularly in extended transients. These assumptions include the following:

1. The reactivity feedback is determined solely by thermal expansions of the grid support plate and load pad region, with all regions having the same thermal expansion coefficient.
2. The displacement of the core midplane is sufficient to estimate the reactivity feedback from radial core expansion.
3. All of the subassembly load pads are in contact throughout the transients.

With these assumptions, the model does not explicitly account for subassembly bowing, and is not capable of calculating changes in core loading conditions during the course of a transient. This deficiency becomes especially important for the extended transients typically encountered for unprotected accidents.

In order to provide a more mechanistic approach to the calculation of the radial core expansion reactivity feedback, and to provide a framework for more detailed modeling as required, the following detailed model was developed. It is intended to overcome the restrictions associated with the assumptions listed above, and to allow a more appropriate use of results from detailed computer code simulations of core behavior, such as those obtained with NUBOW-3D [4-5].

4.5.6.2.1 Model Description

The approach taken in the development of the detailed model is to relate the reactivity feedback from radial core expansion to a change in the size of the core, in the same manner as a uniform dilation of the core is used to calculate the reactivity effect of a change in effective core radius. However, rather than maintain the cylindrical shape associated with a uniform core dilation, an axial profile of core radius is calculated. During a transient, the changes in the axial profile are used in conjunction with a worth curve for radial core expansion to yield the reactivity feedback.

The axial profile of the core radius is obtained from the behavior of an average subassembly in the outer row of the core. The shape of this subassembly is determined by the relative location of the grid plate, the core, the load pads, and any core restraint rings. The shape is also affected by the thermal gradient across the subassembly, as this introduces additional bending of the subassembly. The subassembly is treated as a continuous beam subject to these conditions and restraints. The basic equation that describes these deflections is the differential equation of the elastic curve of the beam,

$$\text{EI}\frac{\text{d}^2\text{y}}{\text{dx}^2} = M_{\text{x}} \tag{4.5-30}$$

where

E = modulus of elasticity, N/m^2

I = moment of inertia of the beam cross-sectional area, m^4

M = bending moment, N-m

x = distance along the beam, m

y = distance perpendicular to the beam, m

This equation is solved subject to various loads and moments, depending on the state of the core. However, since only the displacement is needed, and the forces and moments are never evaluated, the solution is not dependent on the value of EI.

At present, the model is only applicable to the "limited free bow" type of restraint. For this type of restraint, there are load pads just above the top of the core (ACLP) and at the top of the subassembly (TLP). There is also a restraint ring (RR), or core former, around the core at the top load pad elevation. This restraint ring limits the outward motion of the top of the subassembly. With this type of restraint, the shape of the subassembly is determined by one of the following possibilities:

A. Grid Plate/Subassembly Nozzle Clearances Not Exceeded

A.1. No contact at ACLP, RR, or TLP

$$y(x) = \frac{M_1}{6\mathrm{EI}(a - x_1)}(x - x_1)^3 \quad \text{for } x_1 \le x \le a \tag{4.5-31}$$

A.2. No contact at ACLP or TLP; contact at RR

$$y(x) = \frac{M_1}{6\mathrm{EI}(a - x_1)}(x - x_1)^3 + S_{\mathrm{GR}}x \quad \text{for } x_1 \le x \le a \tag{4.5-32}$$

$$S_{\mathrm{GR}} = \frac{R_3 - \frac{M_1}{\mathrm{EI}}\left[\frac{(a-x_1)(L-a)}{2} + \frac{(a-x_1)^2}{6}\right] - \frac{M_2}{\mathrm{EI}}\frac{(L-a)^2}{2}}{L}$$

A.3. No contact at ACLP or RR; contact at TLP

$$y(x) = \frac{M_1}{6\mathrm{EI}(a - x_1)}(x - x_1)^3 + S_{\mathrm{GR}}x \quad \text{for } x_1 \le x \le a \tag{4.5-33}$$

$$S_{\mathrm{GR}} = \frac{R_2 - \frac{M_1}{\mathrm{EI}}\left[\frac{(a-x_1)(L-a)}{2} + \frac{(a-x_1)^2}{6}\right] - \frac{M_2}{\mathrm{EI}}\frac{(L-a)^2}{2}}{L}$$

A.4. Contact at ACLP; no contact at TLP or RR

$$y(x) = \frac{M_1}{6\mathrm{EI}(a - x_1)}(x - x_1)^3 + S_{\mathrm{GR}}x \quad \text{for } x_1 \le x \le a \tag{4.5-34}$$

$$S_{\mathrm{GR}} = \frac{R_1 - \frac{M_1}{\mathrm{EI}}\left[\frac{(a-x_1)^2}{6}\right]}{a}$$

A.5. Contact at ACLP and RR; no contact at TLP

$$y(x) = \frac{M_1}{6\mathrm{EI}(a - x_1)}(x - x_1)^3 - \frac{V_{\mathrm{GR}}}{\mathrm{EI}}\frac{x^3}{6} - \frac{C_1}{\mathrm{EI}}x \quad \text{for } x_1 \le x \le a \tag{4.5-35}$$

$$\frac{V_{\mathrm{GR}}}{\mathrm{EI}} = \frac{P}{\mathrm{EI}}\left[1 - \frac{a}{L}\right]$$

$$\frac{P}{\mathrm{EI}} = \frac{\frac{R_1 L}{a} - R_3 + \frac{M_1}{6\mathrm{EI}}\left(2a^2 - ax_1 - x_1^2\right)\left(\frac{L}{a} - 1\right) + \frac{M_2}{\mathrm{EI}}\frac{(L-a)^2}{2}}{\frac{a}{3}(L-a)^2}$$

$$\frac{C_1}{\mathrm{EI}} = \frac{C_3}{\mathrm{EI}} + \frac{M_1}{\mathrm{EI}}\frac{x_1^2}{2(a - x_1)}$$

$$\frac{C_3}{\mathrm{EI}} = -\frac{R_1}{a} + \frac{M_1}{\mathrm{EI}}\frac{1}{a - x_1}\left[\frac{a^2}{6} - \frac{x_1 a}{2} + \frac{x_1^3}{6a}\right] - \frac{V_{\mathrm{GR}}}{\mathrm{EI}}\frac{a^2}{6}$$

B. Grid Plate/Subassembly Nozzle Clearances Exceeded

B.1. No contact at ACLP; contact at RR

$$y(x) = \frac{M_1}{6\text{EI}(a-x_1)}(x-x_1)^3 + S_{\text{GRMAX}}x - \frac{V_{\text{GR}}}{\text{EI}}\frac{x^3}{6} - \frac{M_{\text{GR}}}{\text{EI}}\frac{x^2}{2} \quad \text{for } x_1 \le x \le a \tag{4.5-36}$$

$$\frac{M_{\text{GR}}}{\text{EI}} = -\frac{V_{\text{GR}}}{\text{EI}}L$$

$$\frac{V_{\text{GR}}}{\text{EI}} = 3\frac{R_3}{L^3} - \frac{M_1}{\text{EI}}\left[\frac{3(a-x_1)(L-a)+(a-x_1)^2}{2L^3}\right] - \frac{M_2}{\text{EI}}\left[\frac{3(L-a)^2}{2L^3}\right] - 3\frac{S_{\text{GRMAX}}}{L^2}$$

B.2. No Contact at ACLP or RR; contact at TLP

$$y(x) = \frac{M_1}{6\text{EI}(a-x_1)}(x-x_1)^3 + S_{\text{GRMAX}}x - \frac{V_{\text{GR}}}{\text{EI}}\frac{x^3}{6} - \frac{M_{\text{GR}}}{\text{EI}}\frac{x^2}{2} \quad \text{for } x_1 \le x \le a \tag{4.5-37}$$

$$\frac{M_{\text{GR}}}{\text{EI}} = -\frac{V_{\text{GR}}}{\text{EI}}L$$

$$\frac{V_{\text{GR}}}{\text{EI}} = 3\frac{R_2}{L^3} - \frac{M_1}{\text{EI}}\left[\frac{3(a-x_1)(L-a)+(a-x_1)^2}{2L^3}\right] - \frac{M_2}{\text{EI}}\left[\frac{3(L-a)^2}{2L^3}\right] - 3\frac{S_{\text{GRMAX}}}{L^2}$$

B.3. Contact at ACLP; no contact at TLP or RR

$$y(x) = \frac{M_1}{6\text{EI}(a-x_1)}(x-x_1)^3 + S_{\text{GRMAX}}x - \frac{V_{\text{GR}}}{\text{EI}}\frac{x^3}{6} - \frac{M_{\text{GR}}}{\text{EI}}\frac{x^2}{2} \quad \text{for } x_1 \le x \le a \tag{4.5-38}$$

$$\frac{M_{\text{GR}}}{\text{EI}} = -\frac{P}{\text{EI}}a$$

$$\frac{V_{\text{GR}}}{\text{EI}} = \frac{P}{\text{EI}}$$

$$\frac{P}{\text{EI}} = 3\frac{R_1}{a^3} - \frac{M_1}{\text{EI}}\frac{(a-x_1)^2}{2a^3} - 3\frac{S_{\text{GRMAX}}}{a^2}$$

B.4. Contact at ACLP and RR; no contact at TLP

$$y(x) = \frac{M_1}{6\text{EI}(a-x_1)}(x-x_1)^3 + S_{\text{GRMAX}}x - \frac{V_{\text{GR}}}{\text{EI}}\frac{x^3}{6} - \frac{M_{\text{GR}}}{\text{EI}}\frac{x^2}{2} \quad \text{for } x_1 \le x \le a \tag{4.5-39}$$

$$\frac{M_{\text{GR}}}{\text{EI}} = \frac{P}{\text{EI}}(L-a) - \frac{V_{\text{GR}}}{\text{EI}}L$$

$$\frac{V_{\text{GR}}}{\text{EI}} = \frac{P}{\text{EI}}\left[1 + \frac{a^3}{2L^3} - \frac{3a^2}{2L^2}\right] + \frac{3R_3}{L^3} - \frac{M_1}{\text{EI}}\frac{\left[3(a-x_1)(L-a)+(a-x_1)^2\right]}{2L^3} - \frac{M_2}{\text{EI}}\frac{\left[3(L-a)^2\right]}{2L^3} - \frac{3S_{\text{GRMAX}}}{L^2}$$

$$\frac{P}{\text{EI}} = \left[R_3\frac{\left(\frac{3a^2}{L^2} - \frac{a^3}{L^3}\right)}{2} - R_1 - \frac{M_1}{\text{EI}}\left[\left(\frac{3a^2}{L^2} - \frac{a^3}{L^3}\right)\frac{(L-a)}{4} + \left(\frac{a^2}{4L^2} - \frac{a^3}{12L^3} - \frac{1}{6}\right)(a-x_1)\right](a-x_1)\right.$$
$$\left. - \frac{M_2}{\text{EI}}\left(\frac{3a^2}{L^2} - \frac{a^3}{L^3}\right)\frac{(L-a)^2}{4} + S_{\text{GRMAX}}\left(a + \frac{a^3}{L^2} - \frac{3a^2}{2L}\right)\right] \Big/ a^3\left(\frac{a^3}{12L^3} - \frac{a^2}{2L^2} + \frac{3a}{4L} - \frac{1}{3}\right)$$

B.5. Contact at ACLP and TLP; no contact at RR

$$y(x) = \frac{M_1}{6\text{EI}(a-x_1)}(x-x_1)^3 + S_{\text{GRMAX}}x - \frac{V_{\text{GR}}}{\text{EI}}\frac{x^3}{6} - \frac{M_{\text{GR}}}{\text{EI}}\frac{x^2}{2} \quad \text{for } x_1 \le x \le a \tag{4.5-40}$$

$$\frac{M_{\text{GR}}}{\text{EI}} = \frac{P}{\text{EI}}(L-a) - \frac{V_{\text{GR}}}{\text{EI}}L$$

$$\frac{V_{\text{GR}}}{\text{EI}} = \frac{P}{\text{EI}}\left[1 + \frac{a^3}{2L^3} - \frac{3a^2}{2L^2}\right] + \frac{3R_2}{L^3} - \frac{M_1}{\text{EI}}\frac{\left[3(a-x_1)(L-a) + (a-x_1)^2\right]}{2L^3} - \frac{M_2}{\text{EI}}\frac{\left[3(L-a)^2\right]}{2L^3} - \frac{3S_{\text{GRMAX}}}{L^2}$$

$$\frac{P}{\text{EI}} = \left[R_2\frac{\left(\frac{3a^2}{L^2} - \frac{a^3}{L^3}\right)}{2} - R_1 - \frac{M_1}{\text{EI}}\left[\left(\frac{3a^2}{L^2} - \frac{a^3}{L^3}\right)\frac{(L-a)}{4} + \left(\frac{a^2}{4L^2} - \frac{a^3}{12L^3} - \frac{1}{6}\right)(a-x_1)\right](a-x_1)\right.$$
$$\left. - \frac{M_2}{\text{EI}}\left(\frac{3a^2}{L^2} - \frac{a^3}{L^3}\right)\frac{(L-a)^2}{4} + S_{\text{GRMAX}}\left(a + \frac{a^3}{L^2} - \frac{3a^2}{2L}\right)\right] \Big/ a^3\left(\frac{a^3}{12L^3} - \frac{a^2}{2L^2} + \frac{3a}{4L} - \frac{1}{3}\right)$$

where

y = radial displacement with respect to the core radius at the grid plate

R_1 = minimum core radius at the above-core load pad with respect to the core radius at the grid plate

R_2 = minimum core radius at the top load pad with respect to the core radius at the grid plate

R_3 = maximum core radius at the restraint ring with respect to the core radius at the grid plate

x = axial elevation

x_1 = elevation of the lower axial blanket/lower reflector interface

a = elevation of the above core load pad

L = elevation of the top load pad

S_{GR} = subassembly slope with respect to vertical at the grid plate

S_{GRMAX} = maximum subassembly slope with respect to vertical at the grid plate

$\frac{M_1}{\text{EI}}$ = thermally induced bending moment in the core region

$\frac{M_2}{\text{EI}}$ = thermally induced bending moment in the above core region

V_{GR} = radial reaction at the grid plate

M_{GR} = applied moment at the grid plate

P = radial force at the above core load pad

EI = modulus of elasticity times the moment of inertia of the subassembly cross-sectional area

The use of the word contact in this context implies that either the outward motion is sufficient for the restraint ring to apply a force preventing further outward motion, or there is sufficient inward motion such that all of the intra-subassembly gaps in the load pad region(s) are eliminated, thus generating a force preventing further inward motion. A grid plate/subassembly nozzle clearance is required for the replacement of subassemblies, and results in a corresponding maximum possible deviation of the subassembly from vertical at the grid plate. When this clearance is exceeded, a moment is applied to the subassembly at the nozzle.

The subassembly is also subjected to a bending moment related to the temperature difference of opposite hex can walls within the subassembly. This temperature difference is converted into an equivalent bending moment, as described in the following section. The temperature difference increases linearly through the core region, from the lower axial blanket to the upper axial blanket. In the upper subassembly region, the temperature difference is assumed to be constant from the upper axial blanket to the top of the subassembly, with the value varying with time as the transient progresses.

The major assumptions incorporated in this model at present include the uniform distribution of core material in the radial direction at every axial elevation and the completely rigid subassembly load pads. Distributing

the material uniformly at each axial elevation as the core radius changes is the same assumption used for the uniform core dilation calculation to obtain the radial core expansion reactivity feedback coefficient, and implies that all of the subassemblies are moving in proportion to their distance from the center of the core. The radial expansion worth gradient and the intra-subassembly temperature gradients tend to be greatest at the edge of the active core, with the result that most of the reactivity feedback effect comes from movement of the outer row of subassemblies. Any movement in the central region of the core that is not proportional to the distance from the center of the core is expected to cause a minor effect. The accuracy of this assumption for any specific reactor core can be evaluated by comparison with results from NUBOW-3D [4-5].

The use of completely rigid subassembly load pads provides slightly greater expansion of the core during certain events as in an unprotected loss-of-flow, and less for other accidents, such as an unprotected loss-of-heat-sink transient. The error introduced by using this assumption is on the order of 15% to 20%, and can be design dependent. The incorporation of deformable load pads and "bridging" of subassemblies may be desirable, and is being considered for a future version of this model. The accuracy of this assumption can also be checked by comparison with NUBOW-3D.

Since the expression given by Eq. (4.5-30) is solved for the various core loading possibilities listed above, the resulting algebraic formulas are incorporated in the code. This avoids the need for a finite difference solution of Eq. (4.5-30), simplifying the computer coding and providing a rapid calculation of the core shape.

4.5.6.2.2 Code Description and Input Requirements

This section contains a description of the algorithm used for calculating the appropriate core shape. This calculation is performed at the start of the transient to establish the steady-state core configuration, and for every step during the transient. The use of the detailed radial core expansion reactivity feedback model requires some of the same input as the simple model plus several other variables. These will be discussed as their use occurs.

The optional model is activated by setting | **IRADEX** | = 4 or 5, where a value of ±5 gives a much more detailed printout, while ±4 only gives results in the PSHORT printout. The first step is to calculate the average temperature of the above-core load pad (ACLP) and top load pad (TLP) regions. This is done using Eq. (4.5-29), as in the basic model. The model sets the location of the ACLP, so that **JSTRDX** does not need to be input. The temperature of the grid plate can be given by either the inlet coolant temperature, or by the wall temperature of the compressible volume used to represent the inlet plenum in PRIMAR-4. This option is discussed in detail for the basic model, and is activated by setting **IRADEX** to the appropriate negative value.

The next step is to calculate the equivalent core radius at the grid plate, the ACLP and the TLP. For this calculation, the following input is needed:

Input	Description
NSUBTC	total number of subassemblies in the active core region, including internal blankets and control subassemblies
MTGRD	material used for the grid plate, where 1 = 316 SS 2 = HT-9
MTACLP **MTTLP**	material used for the ACLP and TLP, where 1 = 316 SS 2 = HT-9
PITCHG	subassembly pitch at the grid plate at reference temperature **TR**
PITCHA	flat-to-flat dimension across the ACLP at reference temperature **TR**
PITCHT	flat-to-flat dimension across the TLP at reference temperature **TR**

Using the pitch at the grid plate along with the steady-state inlet temperature, the equivalent radius of the subassemblies in active core is calculated. As part of the calculation, there is a call to subroutine THRMEX that gives the material thermal expansion as a function of temperature for either 316 SS or HT-9. For the load pad regions, a minimum allowable core radius is calculated based on the size of the load pad region when all of the load pads are pushed together and there are no intra-subassembly gaps. This is possible since the model assumes the load pads all have the same temperature, as described above.

In addition to these dimensions, there are two other geometric constraints, as follows:

Input	Description
SLLMAX	maximum allowable slope of the subassembly at the grid plate with respect to vertical, based on subassembly nozzle/grid plate clearances and dimensions
TLPRRC	clearance between the top load pads and the restraint ring

The value for **SLLMAX** is calculated from the radial clearances of the subassembly nozzle/grid plate socket connection and the length of the connection. The maximum tilt of the subassembly occurs when the maximum radial motion of the nozzle is used, usually inward at the bottom of the nozzle and outward at the top. This number is design-dependent and can vary greatly, even when the subassembly sizes are comparable. The clearance between the top load pad region and the restraint ring is determined by the maximum clearance that would occur between the subassemblies in the outer row of active core and the first row of radial blankets when all of the core subassembly load pads are pushed inward together and all of the radial blanket load pads are pushed outward against the restraint ring. The top load pad/restraint ring clearance is kept constant throughout the transient, i.e. the restraint ring expands as the top load pads expand thermally. This approximation tends to be conservative. Default values for these two input variables have been provided for cases where such detailed design information is not available. Design information should be used wherever possible, as the results can be especially sensitive to the value for **SLLMAX**.

The only other input variable required for determining the core shape are those related to the thermally-induced bending moment:

Input	Description
BNDMM1	applied bending moment at the top of the core region, representing the flat-to-flat temperature difference in the radial direction for the subassemblies at the outer edge of the active core
BNDMM2	applied bending moment in the region above the core, representing the flat-to-flat temperature difference in the radial direction in this region for the subassemblies at the outer edge of active core

The data on the temperature difference must be obtained from a code which performs detailed calculations of the steady-state subassembly temperatures with intersubassembly heat transfer, such as SUPERENERGY-2 [4-6]. Default values are included if such information is not available. The input variables can then be calculated using Eq. (4.5-41).

$$\text{BNDMM1} = \frac{\alpha \Delta T}{D} \tag{4.5-41}$$

where

α = mean thermal expansion coefficient of the subassembly hexcan, $1/K$

ΔT = flat-to-flat temperature difference, K

D = hexcan flat-to-flat dimension, m

The model uses a linear variation in bending moment through the core region, from zero at the bottom of the core to **BNDMM1** at the top. The bending moment **BNDMM2** is applied uniformly from the top of the core to the top of the subassembly. In the transient, the bending moments are modified in proportion to the power-to-flow ratio changes.

Once all of these conditions have been calculated, the algorithm goes through a series of logic to determine the correct combination of forces and moments, as given above. The subassemblies are assumed to be vertical at the grid plate unless there are forces at the ACLP or TLP, or both, which would cause the subassembly to tilt. With the appropriate choice, the algebraic equation corresponding to that loading condition is evaluated for every axial node in the core region. In the printout, the algebraic equation selected is indicated by "CORE SHAPE MODEL =", where the value printed corresponds to the particular case as listed:

CORE SHAPE MODEL

	Steady-State	Transient
Grid plate/subassembly nozzle clearances not exceeded		
No contact at ACLP, RR or TLP	1.0	21.0
No contact at ACLP; contact at RR	2.0	22.0
No contact at ACLP; contact at TLP	3.0	23.0
Contact at ACLP; no contact at TLP or RR	4.0	24.0
Contact at ACLP and RR	5.0	25.0
Grid plate/subassembly nozzle clearances exceeded		
No contact at ACLP; contact at RR	8.0	28.0
No contact at ACLP; contact at TLP	9.0	29.0
Contact at ACLP; no contact at TLP or RR	10.0	30.0
Contact at ACLP and RR	11.0	31.0
Contact at ACLP and TLP	12.0	32.0

In the steady-state, the axial profile of core radius is stored for comparison during the transient. For each step during the transient, the process is repeated and the difference in core radius at each elevation is calculated.

The reactivity worth curve is based on the radial expansion coefficient for a uniform core dilation,

Input	Description
RDEXCF	radial expansion coefficient for a uniform core dilation, $/m

This coefficient is then proportioned among the axial fuel nodes according to the axial power shape. The resulting worth curve provides the radial displacement worth for each axial node in the core. When used in combination with the deflections from steady-state described above, the reactivity feedback from each axial node is determined and the total reactivity feedback from radial core expansion is calculated by summing over the axial fuel nodes. As stated above, this is a very rapid calculation due to the use of algebraic expressions for the subassembly shape, which are the solutions given in Eq. (4.5-31) for the various combinations of force and moments.

4.5.7 Control Rod Drive Expansion Feedback Reactivity

For the control rod drive feedback model, it is assumed that the control rod drives are washed by the outlet coolant from the core. Thermal expansion of the drives due to a rise in core outlet temperature will cause the control rods to be inserted further into the core, providing a negative reactivity component. On the other hand, if the control rod drives are supported on the vessel head, and if the core is supported by the vessel walls, then heating the vessel walls will either lower the core or raise the control rod drive supports, leading to a positive reactivity component. Both the control drive expansion and the vessel wall expansion are accounted for in SAS4A/SASSYS-1. This model is invoked with input variable **ICREXP**.

A simple one-node treatment is used for calculating the temperature of the control rod drives. The equation used is

$$M_{\text{cr}} C_{\text{cr}} \frac{\text{dT}_{\text{cr}}}{\text{dt}} = h_{\text{cr}} A_{\text{cr}} \left(T_{\text{ui}} - T_{\text{cr}} \right) \tag{4.5-42}$$

where

M_{cr} = mass of the control rod drivelines, kg

C_{cr} = specific heat of the rod drivelines, J/kg-K

T_{cr} = control rod drive temperature, K

T_{ui} = coolant temperature in the upper internal structure region, K

h_{cr} = heat transfer coefficient between the coolant and the control rod drive, W/m^2-K

A_{cr} = heat transfer area between the coolant and the control rod drive, m^2

t = time, s

The product of the control rod driveline mass and specific heat is entered as input variable **CRDMC**, and the product of the driveline heat transfer coefficient and area is entered as **CRDHA**.

The coolant temperature in the upper internal structure region is calculated using

$$\frac{\text{dT}_{\text{ui}}}{\text{dt}} = w_{\text{c}} \frac{T_{\text{mm}} - T_{\text{ui}}}{\rho_{\text{u}} V_{\text{ui}}} \tag{4.5-43}$$

where

w_{c} = core outlet flow rate

T_{mm} = mixed mean coolant outlet temperature

ρ_{u} = sodium density in the outlet plenum

V_{ui} = coolant volume in the upper internal structure region

Initially both T_{ui} and T_{cr} are set equal to the steady-state mixed mean outlet temperature. The UIS volume, V_{ui} is entered as input variable **UIVOL**.

During the transient calculation, Eq. (4.5-43) is approximated with

$$T_{\text{ui}} \left(t + \Delta t \right) = \frac{T_{\text{ui}} \left(t \right) + x T_{\text{mm}} \left(t + \Delta t \right)}{1 + x} \tag{4.5-44}$$

where

$$x = \frac{w_{\text{c}} \left(t + \Delta t \right) \Delta t}{\rho_{\text{u}} V_{\text{ui}}} \tag{4.5-45}$$

For this calculation, only channels with positive outlet flow rates contribute to w_c and T_{mm}. Eq. (4.5-42) is approximated as

$$M_{cr}C_{cr}\frac{T_{cr}(t+\Delta t)-T_{cr}(t)}{\Delta t}=h_{cr}A_{cr}\left[T_{ui}(t+\Delta t)-T_{cr}(t+\Delta t)\right] \tag{4.5-46}$$

or

$$T_{cr}(t+\Delta t)=\frac{T_{cr}(t)+d\,T_{ui}(t+\Delta t)}{1+d} \tag{4.5-47}$$

where

$$d=\frac{h_{cr}A_{cr}}{M_{cr}C_{cr}}\Delta t \tag{4.5-48}$$

The axial expansion of the control rod drive, Δz_{cr}, is calculated as

$$\Delta z_{cr}(t)=L_{cr}\alpha_{cr}\left[T_{cr}(t)-T_{cr}(0)\right] \tag{4.5-49}$$

where L_{cr} is the length of the control rod drive washed by the outlet sodium, and α_{cr} is the thermal expansion coefficient. These data are entered as **CRDLEN** and **CRDEXP**.

The vessel wall expansion is calculated on the basis of the temperatures calculated by PRIMAR-4 for the walls of the liquid elements or compressible volumes that represent the vessel wall. For a typical pool reactor, the vessel wall would be the wall of the cold pool; but for some reactor designs a number of compressible volumes and liquid elements would be used to represent the vessel wall. The expansion of the vessel wall, Δz_v, is calculated as

$$\Delta z_v=\sum_k\left[\overline{T}_k(t)-\overline{T}_k(0)\right]L_k\alpha_k \tag{4.5-50}$$

where

$\overline{T}_k$ = average wall temperature of the k-th compressible volume or liquid element in the vessel wall

L_k = length of the vessel wall represented by the k-th compressible volume or element

α_k = thermal expansion coefficient of the vessel wall

The net movement, Δz_n, is calculated as

$$\Delta z_n=\Delta z_{cr}-\Delta z_v \tag{4.5-51}$$

and the reactivity feedback, δk_{cr}, is calculated as

$$\delta k_{cr}=a_{cr}\Delta z_n+b_{cr}(\Delta z_n)^2 \tag{4.5-52}$$

where a_{cr} and b_{cr} are user-supplied coefficients entered as **ACRDEX** and **BCRDEX**.

A multiple-node version of this model is in development, but has not been verified for production use.

4.5.8 Fuel and Cladding Relocation Feedback Reactivity

The fuel and cladding relocation reactivity feedbacks are calculated as the product of the input material reactivity worth (**CLADRA** and **FUELRA**) and the change in the axial material mass distribution since the initial steady-state condition. Symbolically this is represented by

$$\delta k(t)=\sum_i\sum_j\left(\frac{\Delta k}{\Delta m}\right)_{ij}\left[m_{ij}(t)-m_{ij}(0)\right]$$

where $\left(\frac{\Delta k}{\Delta m}\right)_{ij}$ is the material reactivity worth in axial node j of channel i, $m_{ij}(t)$ is the material mass at axial node j in channel i at time t, and $m_{ij}(0)$ is the initial steady-state node material mass. The input worth curves may be input on the fuel (MZ) mesh, or the coolant (MZC) mesh according to the input value of **IREACZ**. The initial and transient axial mass distributions are computed internally from input design geometry and density data, and the fuel and cladding relocation models and solutions.

4.5.9 EBR-II Reactivity Feedback model

A reactor-specific set of reactivity feedback correlations has been implemented in SAS4A/SASSYS-1 for analysis of the EBR-II reactor and plant. The formulation of these correlations is based on the reactivity feedback model used in the NATDEMO computer program [4-7, 4-8], which is based on analysis of reactivity temperature coefficients in EBR-II Run 93 [4-9]. This documentation of the SAS4A/SASSYS-1 EBR-II reactivity feedback model is taken from notes provided by White [4-10] and Herzog [4-11].

In SAS4A/SASSYS-1, the EBR-II reactivity feedback is assumed to be composed of nine components. These components are: 1) fuel expansion, 2) coolant expansion, 3) stainless steel expansion, 4) axial reflector sodium expansion, 5) radial reflector sodium expansion, 6) fuel Doppler effect, 7) control rod bank expansion, 8) upper grid plate expansion, and 9) core subassembly bowing.

Each of these effects will be considered independently and the method in which the feedback magnitude is determined will be given. The source and nature (linear or nonlinear) of the term will also be discussed.

The reactivity in a steady state critical reactor is defined as zero. In the EBR-II feedback calculation, a parameter, which shall be named $\zeta(0)$, is defined at time zero (this treatment is not used in the other reactivity calculations performed by SAS4A/SASSYS-1). At later times, $\zeta(t)$ is calculated. The difference, $\delta k(t) = \zeta(t) - \zeta(0)$, is the reactivity introduced from feedback effects at time t.

4.5.9.1 Fuel Expansion

Both radial and axial expansion of the fuel are considered to be linear terms.

4.5.9.1.1 Axial Expansion

The method used is derived from Ref. 4-12. It has been modified to include the possibility that contact can occur between the fuel and cladding, thus altering the expression for the amount of expansion. Correlations from the Metallic Fuels Handbook [4-13] are used to evaluate the linear expansion coefficient and Young's modulus for the fuel and cladding types used in EBR-II fuel elements. First, the following correlation is used to determine the coefficient of linear expansion for U-5FS fuel:

$$\alpha_{\mathrm{LF}} = \begin{cases} 1.264 \times 10^{-5} - 1.7964 \times 10^{-9}\overline{T} + 2.0532 \times 10^{-11}\overline{T}^2 & \overline{T} < 941 \\ 1.73 \times 10^{-5} & 941 < \overline{T} < 1048 \\ 1.775 \times 10^{-5} + 8.761 \times 10^{-9}\overline{T} - 3.717 \times 10^{-12}\overline{T}^2 & 1048 < \overline{T} < 1480 \\ 2.55 \times 10^{-5} & 1480 < \overline{T} \end{cases}$$

and for U-10Zr fuel, the following correlation is used:

$$\alpha_{\mathrm{LF}} = \begin{cases} 1.658 \times 10^{-5} - 2.104 \times 10^{-8}\overline{T} + 3.345 \times 10^{-11}\overline{T}^2 & \overline{T} < 900 \\ 2.25 \times 10^{-5} & 900 < \overline{T} \end{cases}$$

and for U-10Zr-20Pu fuel, the following correlation is used:

$$\alpha_{\mathrm{LF}} = \begin{cases} 1.73 \times 10^{-5} & \overline{T} < 868 \\ 1.98 \times 10^{-5} & 868 < \overline{T} \end{cases}$$

The $\overline{T}$ used here is the mass-average temperature for the fuel in a particular channel for a particular axial layer.

For SS316 or D-9 cladding, the steel linear thermal expansion coefficient is calculated from:

$$\alpha_{\mathrm{Lss}} = 5.189 \times 10^{-5} - 6.4375 \times 10^{-4}\overline{T}^{-\frac{1}{2}} - 1.00862 \times 10^{-8}\overline{T}$$

and for HT-9 cladding the linear thermal expansion coefficient is calculated from:

$$\alpha_{\mathrm{Lss}} = 1.62307 \times 10^{-6} + 2.84714 \times 10^{-8}\overline{T} - 1.65103 \times 10^{-11}\overline{T}^{2}$$

where $\overline{T}$ is the mass-average cladding temperature at a particular axial location in a particular channel.

Correlations [4-13] for the Young's modulus of metal fuels are functions of the fuel temperature and porosity **PRSTY**. The following correlation is used to determine the Young's modulus of U-5Fs fuel:

$$Y_{\mathrm{f}} = \begin{cases} 1.5123 \times 10^{11}\,(1 - 1.2P)\left(1 - 1.06\frac{\overline{T}-588}{1405}\right) & \overline{T} < 923 \\ 1.5123 \times 10^{-11}\,(1 - 1.2P)\left(1 - 1.06\frac{\overline{T}-588}{1405}\right) - 0.3Y_{\mathrm{f}}\,(923) & 923 < \overline{T} \end{cases}$$

For U-10Zr fuel, the following correlation is used:

$$Y_{\mathrm{f}} = \begin{cases} 1.4349 \times 10^{11}\,(1 - 1.2P)\left(1 - 1.06\frac{\overline{T}-588}{1405}\right) & \overline{T} < 923 \\ 1.4349 \times 10^{-11}\,(1 - 1.2P)\left(1 - 1.06\frac{\overline{T}-588}{1405}\right) - 0.3Y_{f}\,(923) & 923 < \overline{T} \end{cases}$$

And for U-10Zr-20 Pu fuel, the following correlation is used:

$$Y_{\mathrm{f}} = \begin{cases} 1.1149 \times 10^{11}\,(1 - 1.2P)\left(1 - 1.06\frac{\overline{T}-588}{1405}\right) & \overline{T} < 923 \\ 1.1149 \times 10^{-11}\,(1 - 1.2P)\left(1 - 1.06\frac{\overline{T}-588}{1405}\right) - 0.3Y_{\mathrm{f}}\,(923) & 923 < \overline{T} \end{cases}$$

For SS316 or D-9 cladding, the steel Young's modulus is calculated from:

$$Y_{\mathrm{ss}} = 2.01 \times 10^{11} - 7.929 \times 10^{7}\overline{T}$$

and for HT-9 cladding the Young's modulus is calculated from:

$$Y_{\mathrm{ss}} = 2.137 \times 10^{11} - 1.0274 \times 10^{8}\overline{T}$$

The expression for the expansion in the fuel is

$$\delta_{\mathrm{f}} = \alpha_{\mathrm{Lf}}\Delta T_{\mathrm{f}}$$

for no contact (fuel burnup < 2.9%), and

$$\delta_{\mathrm{f}} = \frac{\alpha_{\mathrm{Lss}}\Delta T_{\mathrm{ss}} Y_{\mathrm{ss}} A_{\mathrm{ss}} + \alpha_{\mathrm{Lf}}\Delta T_{\mathrm{f}} Y_{\mathrm{f}} A_{\mathrm{f}}}{Y_{\mathrm{ss}} A_{\mathrm{ss}} + Y_{\mathrm{f}} A_{\mathrm{f}}}$$

for contact (fuel burnup > 2.9%), where α_{Lf} and α_{Lss} are the thermal expansion coefficients for the fuel and stainless steel, ΔT_{f} and ΔT_{ss} are the temperature changes for the fuel and stainless steel, Y_{f} and Y_{ss} are Young's modulus for the fuel and stainless steel, and A_{f} and A_{ss} are the cross sectional areas of the fuel and stainless steel.

As described above, the change in temperature is not calculated directly by the code. Instead, the code expands the expression into two terms: one involving steady state conditions, $\zeta(0)$, and the second involving the conditions at some time t, $\zeta(t)$. The reactivity feedback is determined using the expression $\delta k(t) = \zeta(t) - \zeta(0)$. The parameter ζ due to the expansion can be expressed as:

$$\zeta_{\text{fa}} = -\left[\left(\frac{\partial \text{k}}{\frac{\partial \text{N}}{N}}\right)_{\text{f}} - \left(\frac{\partial \text{k}}{\frac{\partial \text{H}}{H}}\right)_{\text{f}}\right] \frac{\alpha_{\text{Lss}} \overline{T}_{\text{ss}} Y_{\text{ss}} A_{\text{ss}} + \alpha_{\text{Lf}} \overline{T}_{\text{f}} Y_{\text{f}} A_{\text{f}}}{\left(Y_{\text{ss}} A_{\text{ss}} + Y_{\text{f}} A_{\text{f}}\right) \beta}$$

where β is the delayed neutron fraction, ζ_{fa} is the reactivity due to fuel axial expansion in dollars, $\left(\frac{\partial \text{k}}{\frac{\partial \text{N}}{N}}\right)_{\text{f}}$ is the change in k with a relative change in the number density of the fuel, and $\left(\frac{\partial \text{k}}{\frac{\partial \text{H}}{H}}\right)_{\text{f}}$ is the change in k with a relative change in the height of the fuel, for the case when the clad and fuel are in contact and

$$\zeta_{\text{fa}} = -\left[\left(\frac{\partial \text{k}}{\frac{\partial \text{N}}{N}}\right)_{\text{f}} - \left(\frac{\partial \text{k}}{\frac{\partial \text{H}}{H}}\right)_{\text{f}}\right] \frac{\alpha_{\text{Lf}} \overline{T}_{\text{f}}}{\beta}$$

for the case where there is no contact. The two partial derivatives in both of these expressions are entered as input to SAS4A/SASSYS-1 as input variables **YKNF** and **YKHF**.

Because some channels will have both assemblies in which the cladding and fuel do contact and assemblies in which they do not contact, the code uses the equation:

$$\begin{aligned}\zeta_{\text{fa}} = -\left[\left(\frac{\partial \text{k}}{\frac{\partial \text{N}}{N}}\right)_{\text{f}} - \left(\frac{\partial \text{k}}{\frac{\partial \text{H}}{H}}\right)_{\text{f}}\right] & \frac{\alpha_{\text{Lss}} \overline{T}_{\text{ss}} Y_{\text{ss}} A_{\text{ss}} + \alpha_{\text{Lf}} \overline{T}_{\text{f}} Y_{\text{f}} A_{\text{f}}}{\left(Y_{\text{ss}} A_{\text{ss}} + Y_{\text{f}} A_{\text{f}}\right) \beta} \left(1 - F_{\text{LowBU}}\right) \\ & - \left[\left(\frac{\partial \text{k}}{\frac{\partial \text{N}}{N}}\right)_{\text{f}} - \left(\frac{\partial \text{k}}{\frac{\partial \text{H}}{H}}\right)_{\text{f}}\right] \frac{\alpha_{\text{Lf}} \overline{T}_{\text{f}}}{\beta} F_{\text{LowBU}}\end{aligned}$$

where F_{LowBU} is the fraction of pins with < 2.9% burnup. This fraction is entered to SAS4A/SASSYS-1 as variable **FLOWBU**.

The temperature of the stainless steel (cladding) is determined using a weighted mass-average of the cladding temperature. The cladding is composed of three radial nodes. In the weighting, the middle node is given twice the weight of the two other nodes.

4.5.9.1.2 Radial Expansion

The radial expansion of the fuel contributes to the reactivity principally by displacing the bond gap sodium from the core. This displacement is of the sodium between the cladding and the fuel. Once the fuel reaches a burnup of 2.9% the cladding is in contact with the fuel and there is no additional displacement, and therefore, no additional reactivity change. The amount by which the fuel volume increases can be determined using a two dimensional isotropic approximation, i.e.: fuel volume increase is $V_{\text{f}} 2\alpha_{\text{Lf}}$. The fractional decrease in the sodium volume is then $\frac{V_{\text{f}} 2\alpha_{\text{Lf}}}{V_{\text{Na}}}$. The ζ value associated with this sodium expulsion is:

$$\zeta_{\text{fr}} = \left(-\frac{2V_{\text{f}}\ \alpha_{\text{Lf}}}{V_{\text{Na}}}\right)\left(\frac{\partial \text{k}}{\frac{\partial \text{N}}{N}}\right)_{\text{Na}} \frac{\overline{T}_{\text{f}}}{\beta}$$

when there is no contact and $\zeta_{\text{fr}} = 0$ when there is contact. In these expressions, $\left(\frac{\partial \text{k}}{\frac{\partial \text{N}}{N}}\right)_{\text{Na}}$ is the change in k with respect to a relative change in the number density of the sodium, which is entered as input variable **YKNNA**. Therefore, the value of ζ for the radial expansion of the fuel can be expressed as:

$$\zeta_{\text{fr}} = \left(-\frac{2V_{\text{f}}\alpha_{\text{Lf}}}{V_{\text{Na}}}\right)\left(\frac{\partial \text{k}}{\frac{\partial \text{N}}{N}}\right)_{\text{Na}} \frac{\overline{T}_{\text{f}}}{\beta} F_{\text{LowBU}}$$

The same type of summation as given above is used to volume weight the reactivities (and temperatures, ζ values) from the different axial layers and channels.

4.5.9.2 Coolant Expansion

When the sodium coolant expands, it increases the leakage from the reactor, decreases sodium capture and results in spectral shifts. Sodium expansion is assumed to be a linear effect. The sodium coolant expansion is treated in a manner developed in Ref. 4.9. The equation used is:

$$\zeta_{\mathrm{c}} = \left(\frac{\partial \mathrm{k}}{\frac{\partial \mathrm{N}}{N}} \right) \alpha_{\mathrm{V_{Na}}} \frac{\overline{T}_{\mathrm{Na}}}{\beta}$$

where α_{VNa} is the thermal volumetric expansion coefficient, $\overline{T}_{\mathrm{Na}}$ is the sodium temperature, and the sodium temperature is the value assigned by the code for a particular channel and axial layer. The sodium volumetric thermal expansion coefficient is calculated from the local sodium temperature in the correlation for the sodium volumetric thermal expansion coefficient given in Eq. (12.13-10).

The reactivity worths are summed in the manner described in the fuel axial expansion section. The sodium number density reactivity coefficient is entered as **YKNNA**, the radial sodium reactivity worth factor is entered as **XRNSHP**, and the axial sodium reactivity worth weighting is taken as the normalized axial shape of **VOIDRA**.

4.5.9.3 Stainless Steel Expansion

Both axial and radial steel expansion coefficients are considered to be linear effects.

4.5.9.3.1 Axial Expansion

The axial expansion of the stainless steel cladding results in a decrease in the number density. The amount of expansion, as in the case of the fuel, is different for conditions where there is contact or no contact. In the event of no contact (burnup < 2.9%), the expression for the reactivity is:

$$\zeta_{\mathrm{ssa}} = -\left(\frac{\partial \mathrm{k}}{\frac{\partial \mathrm{N}}{N}} \right)_{\mathrm{ss}} \frac{\alpha_{\mathrm{Lss}} \overline{T}_{\mathrm{ss}}}{\beta}$$

where α_{Lss} is the linear thermal expansion coefficient for the cladding, and $\left(\frac{\partial \mathrm{k}}{\frac{\partial \mathrm{N}}{N}} \right)_{\mathrm{ss}}$ is the change in k with respect to the relative number density in the stainless steel.

The partial derivative above is an input parameter, **YKNSS**. If there is contact, the following expression applies:

$$\zeta_{\mathrm{ssa}} = -\left(\frac{\partial \mathrm{k}}{\frac{\partial \mathrm{N}}{N}} \right)_{\mathrm{ss}} \frac{\alpha_{\mathrm{Lss}} \overline{T}_{\mathrm{ss}} Y_{\mathrm{ss}} A_{\mathrm{ss}} + \alpha_{\mathrm{Lf}} \overline{T}_{\mathrm{f}} Y_{\mathrm{f}} A_{\mathrm{f}}}{\left(Y_{\mathrm{ss}} A_{\mathrm{ss}} + Y_{\mathrm{f}} A_{\mathrm{f}} \right) \beta}$$

The code therefore contains the expression:

$$\zeta_{\mathrm{ssa}} = -\left(\frac{\partial \mathrm{k}}{\frac{\partial \mathrm{N}}{N}} \right)_{\mathrm{ss}} \frac{\alpha_{\mathrm{Lss}} \overline{T}_{\mathrm{ss}}}{\beta} F_{\mathrm{LowBU}} - \left(\frac{\partial \mathrm{k}}{\frac{\partial \mathrm{N}}{N}} \right)_{\mathrm{ss}} \frac{\alpha_{\mathrm{Lss}} \overline{T}_{\mathrm{ss}} Y_{\mathrm{ss}} A_{\mathrm{ss}} + \alpha_{\mathrm{Lf}} \overline{T}_{\mathrm{f}} Y_{\mathrm{f}} A_{\mathrm{f}}}{\left(Y_{\mathrm{ss}} A_{\mathrm{ss}} + Y_{\mathrm{f}} A_{\mathrm{f}} \right) \beta} \left(1 - F_{\mathrm{LowBU}} \right)$$

The radial and axial weighting of the steel expansion reactivity is carried out in the same manner as for the fuel expansion reactivity. The radial steel reactivity worth shape factor is entered as **XRSSHP**, and the axial shape is taken as the normalized axial shape of input array **CLADRA**.

4.5.9.3.2 Radial Expansion

The reactivity feedback due to radial expansion of the cladding results from the displacement of sodium from the core. This removal is independent of the burnup because the expansion is directed toward the coolant channel. The expression for the radial expansion of the stainless steel structure is:

$$\zeta_{\text{ssr}} = -2\frac{V_{\text{ss}}}{V_{\text{Na}}}\left(\frac{\partial \text{k}}{\frac{\partial \text{N}}{N}}\right)_{\text{Na}}\frac{\alpha_{\text{Lss}}\overline{T}_{\text{ss}}}{\beta}$$

where V_{ss} is the volume of stainless steel in the cladding, and ζ_{ssr} is the reactivity feedback due to radial expansion of the clad.

The same type of summation as given above (Section 4.5.9.3.1) is used to volume weight the reactivities.

4.5.9.4 Axial Reflector Sodium Expansion

In the axial reflector the reactivity feedbacks stem mostly from the change in leakage. The upper and lower axial reflector sodium expansion are treated as linear effects.

4.5.9.4.1 Upper Reflector

The expression for the upper reflector sodium parameter ζ is:

$$\zeta_{\text{ur}} = \left(\frac{\partial \text{k}}{\partial \text{T}}\right)_{\text{ar}}\frac{\overline{T}_{\text{ur}}}{\beta}$$

where $\overline{T}_{\text{ur}}$ is the average temperature in the upper reflector, and $\left(\frac{\partial \text{k}}{\partial \text{T}}\right)_{\text{ar}}$ is the change in k with respect to a change in axial reflector temperature.

The input value for the above partial derivative is entered as variable **YRCUR**.

To determine the average temperature of the coolant in the upper reflector, the following equation is used:

$$\overline{T}_{\text{uf}} = \frac{\sum_{\text{i}=1}^{N} V_{\text{i}}T_{\text{i}} + \sum_{\text{j}=1}^{M}\sum_{\text{k}=1}^{L(j)} V_{\text{j,k}}T_{\text{j,k}}}{\sum_{\text{i}=1}^{N} V_{\text{i}} + \sum_{\text{j}=1}^{M}\sum_{\text{k}=1}^{L(j)} V_{\text{j,k}}}$$

where N is the number of axial nodes in the gas plenum, M is the number of zones in the upper reflector, $L\,(j)$ is the number of axial nodes in zone j. Also, V_{i} is the volume of coolant node i in the plenum space, T_{i} is the temperature of the node i coolant in the plenum space, $V_{\text{j,k}}$ is the volume of coolant node k in zone j in the reflector area, and $T_{\text{j,k}}$ is the temperature of the coolant in node k of zone j in the reflector area.

In this analysis, only the sodium coolant is considered to determine the reactivity feedback. This is done for two reasons: the difference between the sodium coolant and stainless steel temperature is small and the relative contributions of the stainless steel and coolant expansions are unknown. The volumes used are that of the flow, not of the structure and flow.

4.5.9.4.2 Lower Reflector

The expression for the lower reflector sodium feedback is:

$$\zeta_{\text{lr}} = \left(\frac{\partial \text{k}}{\partial \text{T}}\right)_{\text{ar}} \frac{\overline{T}_{\text{lr}}}{\beta}$$

where the partial derivative is entered as input variable **YLCLR** and $\overline{T}_{\text{lr}}$ is the average temperature in the lower reflector.

The same temperature volume-weighting scheme used in the upper reflector is also used in the lower reflector. Again, the temperatures are that of the coolant and the volumes are that in which flow occurs.

4.5.9.5 Radial Reflector Expansion

The reactivity change that results from a temperature change in the radial reflector is mainly due to the change in density of the sodium in the reflector, which results in a change in the leakage. The radial reflector expansion is treated as a linear effect. The expression for the radial reflector sodium feedback is:

$$\zeta_{\text{rr}} = \left(\frac{\partial \text{k}}{\partial \text{T}}\right)_{\text{rr}} \frac{\overline{T}_{\text{rr}}}{\beta}$$

where $\left(\frac{\delta \text{k}}{\delta \text{T}}\right)_{\text{rr}}$ is the change in k with respect to the radial reflector temperature, and $\overline{T}_{\text{rr}}$ is the average temperature of the radial reflector.

The partial derivative is entered as input variable **YRCRR**. The nodes from which the average temperature is determined are those in which **LCHTYP** is set to 2, indicating a stainless steel reflector subassembly, and those of the bypass region at the core level. For the channel subassemblies, only zone 5 (currently the core) is used in the volume weighting. For the bypass channel, the temperatures are also volume weighted. The coolant temperature is volume weighted in all axial nodes of all channels to obtain the average temperature. The equation used in the code is:

$$\overline{T}_{\text{rr}} = \frac{\sum_{\text{i}} V_{\text{by,i}} T_{\text{by,i}} + \sum_{\text{j}} V_{\text{ch,j}} T_{\text{ch,j}}}{V_{\text{by}} + V_{\text{ch}}}$$

where $V_{\text{by,i}}$ is the volume of bypass region i, $T_{\text{by,i}}$ is the temperature of the bypass region i, V_{by} is the total volume of the bypass region corresponding to the reflector, $V_{\text{ch,j}}$ is the volume of axial layer j of the stainless steel containing channel, $T_{\text{ch,j}}$ is the temperature of axial layer j of the stainless steel containing channel, V_{ch} is the total volume of the stainless steel channel.

4.5.9.6 Doppler Effect

The Doppler effect is a nonlinear phenomenon that results from the change in effective cross sections at different temperatures. The temperature coefficient for the Doppler effect $\left(\frac{\delta \text{k}}{\delta \text{T}}\right)_{\text{D}}$ is a function of fuel temperature:

$$\left(\frac{\partial \text{k}}{\partial \text{T}}\right)_{\text{D}} = \frac{k_{\text{D}}}{\overline{T}_{\text{f}}}$$

where k_{D} is an input variable **YRCDOP** and $\overline{T}_{\text{f}}$ is the channel average fuel temperature. Since the Doppler temperature coefficient is temperature dependent, an average value is used.

$$\left\langle \left(\frac{\partial \text{k}}{\partial \text{T}}\right)_{\text{D}} \right\rangle = \frac{\int_{\overline{T}_{\text{f}}(0)}^{\overline{T}_{\text{f}}(t)} \left(\frac{\partial \text{k}}{\partial \text{T}}\right)_{\text{D}} \text{d}\overline{T}_{\text{f}}}{\int_{\overline{T}_{\text{f}}(0)}^{\overline{T}_{\text{f}}(t)} \text{d}\overline{T}_{\text{f}}} = \frac{k_{\text{D}}}{\overline{T}_{\text{f}}(t) - \overline{T}_{\text{f}}(0)} \ln\left[\frac{\overline{T}_{\text{f}}(t)}{\overline{T}_{\text{f}}(0)}\right]$$

where $\overline{T}_{\mathrm{f}}(t)$ is the channel average fuel temperature at time t, and $\overline{T}_{\mathrm{f}}(0)$ is the steady-state channel average fuel temperature.

The total feedback δk_{D} due to the Doppler effect is the sum of the Doppler feedbacks for each fueled channel. This is implemented in the code as:

$$\delta k_{\mathrm{D}} = \sum_{\mathrm{i}=1}^{N} \left\langle \left(\frac{\partial \mathrm{k}}{\partial \mathrm{T}} \right)_{\mathrm{D}} \right\rangle_{\mathrm{i}} \frac{V_{\mathrm{i}}}{V_{\mathrm{tot}}} \left[\overline{T}_{\mathrm{f,i}}(t) - \overline{T}_{\mathrm{f,i}}(0) \right] = \frac{k_{\mathrm{D}}}{V_{\mathrm{tot}}} \sum_{\mathrm{i}=1}^{N} V_{\mathrm{i}} \ln \left[\frac{\overline{T}_{\mathrm{f,i}}(t)}{\overline{T}_{\mathrm{f,i}}(0)} \right]$$

where the temperatures given above are currently weighted by the axial and radial reactivity worths.

4.5.9.7 Control Rod Bank Expansions

The change in reactivity that results from control rod bank temperature change is a result of the support of these rods. These are suspended from the top of the reactor. When a temperature change occurs, the rods expand or contract, forcing some of the fuel into or out of the core. This results in a reactivity change. The control rod expansion is treated as a linear effect. Although there are several portions of the control rod that undergo expansion when the temperature is increased, the current code only models two. The first of these is the control rod fuel and the second is the driveline expansion. The expression found in the code is:

$$\zeta_{\mathrm{cr}} = \frac{\left(\frac{\partial \mathrm{k}}{\partial \mathrm{T}}\right)_{\mathrm{cr}}}{\beta} \left[F_{\mathrm{cr}} \frac{\sum_{\mathrm{i}=1}^{N} \overline{T}_{\mathrm{f,i}} V_{\mathrm{i}}}{V_{\mathrm{tot}}} + (1 - F_{\mathrm{cr}}) T_{\mathrm{CV2}} \right]$$

where T_{CV2} is the temperature in the upper plenum (CV#2 in SAS4A/SASSYS-1), $\left(\frac{\delta \mathrm{k}}{\delta \mathrm{T}}\right)_{\mathrm{cr}}$ is the change is the change in k with respect to control rod temperature, and F_{cr} is the fraction of the response due to the channel temperature change.

The partial derivative above is entered as input variable **YRCCR** and F_{cr} is entered as **FCR**.

The model given above differs from previous models [Ref. 4-12] because it does not include different time constants which characterize the different portions of the stainless steel control rod structure; instead, they are treated as the two terms given above; one proportional to fuel temperature and the other proportional to the exit plenum temperature. For steady state or a series of steady states this will not result in any significant error.

4.5.9.8 Upper Grid Plate Expansion

The upper grid plate is located below the core and serves as a support for the hex can. Grid plate expansion impacts the reactivity by displacing the assembly rows in the reactor. The upper grid plate expansion is treated as a linear effect. Reference 4-9 gives the following expression for the feedback:

$$\zeta_{\mathrm{gp}} = \frac{T_{\mathrm{CV1}}(t)}{\beta} \left(\frac{\partial \mathrm{k}}{\partial \mathrm{T}} \right)_{\mathrm{gp}}$$

where ζ_{gp} is the reactivity due to the upper grid plate expansion, $T_{\mathrm{CV1}}(t)$ is the temperature of the inlet plenum (CV#1 in SAS4A/SASSYS-1), and $\left(\frac{\delta \mathrm{k}}{\delta \mathrm{T}}\right)_{\mathrm{gp}}$ is the feedback coefficient associated with the grid plate.

The partial derivative above is an input parameter entered as **YRCGP**.

4.5.9.9 Bowing and Unspecified Parameters

In the EBR-II reactivity feedback model, "bowing" is a "catch all" for terms which can not specifically be identified and therefore may contain terms other than that which result from the bowing of the assembly. True assembly bowing results when one side of a fuel assembly is at a higher temperature than the other. This results in a different expansion of the two sides moving the assembly radially. Bowing contributes to the reactivity by increasing or decreasing the amount of fuel in relatively high worth portions of the core.

The overall "bowing" term in the EBR-II model is treated as a linear effect that is modeled by one line below a threshold and by another line above the threshold. The effect is a function of the normalized temperature rise across the core ΔT_{norm}. For normalized temperatures above the threshold, which is given by the input variable **YTCUT**, the bowing reactivity δk_{bw} is given by:

$$\delta k_{\text{bw}} = A + B\Delta T_{\text{norm}}$$

For normalized temperatures below the threshold, the bowing reactivity δk_{bw} is given by:

$$\delta k_{\text{bw}} = \left(\frac{A}{\text{YTCUT}} + B \right) \Delta T_{\text{norm}}$$

The values for the input coefficients A and B are entered as **YABOW** and **YBBOW**.

The normalized temperature rise is the ratio of the temperature rise across the core at time t to the temperature rise across the core at full power and flow conditions. The temperature rise is modeled as the difference between the upper and lower plena temperatures (found in YTLCV2 and YTLCV1) or as the difference between the average sodium temperature for the fueled channels $\overline{T}_{\text{Na}}$ and the temperature of the lower plenum YTLCV1. Which model is used for the normalized temperature rise is determined by an input parameter **IBOWTP**. For **IBOWTP** equal to zero, the normalized temperature rise is based on the outlet and inlet plena temperatures. In the case, the normalized factor **YDELT0** gives the difference between the outlet and inlet plena temperatures at full power and flow conditions. For **IBOWTP** not equal to zero, the normalized temperature rise is based on the average sodium temperature in fueled channels and the temperature of the inlet plenum. In this case, the normalized factor **YDELT0** gives the difference between the average sodium temperature in the fueled channels and the temperature of the inlet plenum at full power and flow conditions.

4.6 Solution Methods

The SAS4A/SASSYS-1 point kinetics equations are solved using a technique model by Kaganove [4-13] and extended by Fuller [4-14]. The decay-heat model is solved according to the method developed by Woodruff for the SAS3A computer code [4-1]. In general, the SAS4A/SASSYS-1 neutronics solution methods have been adopted from SAS3A.

4.6.1 Point Kinetics

The solution of the point kinetics model equations is carried out assuming that (1) the reactivity has been specified, and (2) the number of delayed-neutron precursors is limited to six or less. This latter assumption requires the interpretation of the precursor yields and decay constants as average values, typical for a particular blend of isotopes.

Given these assumptions, the first step in the solution of the point kinetics equations is the integration of the delayed-neutron precursor energy production balance equations given by Eq. (4.3-1) over a time interval. In

SAS4A/SASSYS-1, this time interval is taken as the main time step. If conditions are known at time t, then integration of Eq. (4.3-1) to time $t + \Delta t$ gives

$$C_{\mathrm{i}}(t+\Delta t) = C_{\mathrm{i}}(t)\, e^{-\lambda_{\mathrm{i}}\Delta t} + \frac{\beta_{\mathrm{i}}}{\Lambda} e^{-\lambda_{\mathrm{i}}(t+\Delta t)} \int_t^{t+\Delta t} \phi(t')\, e^{-\lambda_{\mathrm{i}} t'}\, \mathrm{dt}' \tag{4.6-1}$$

The expressions given by Eq. (4.6-1) for the C_{i} at the advanced time are substituted directly into Eq. (4.2-4). Next, the time variation of the fission power amplitude over the time step is assumed to be

$$\phi(t+\Delta t) = \phi(t) + \phi_1 \Delta t + \phi_2 (\Delta t)^2 \tag{4.6-2}$$

where ϕ_1 and ϕ_2 are constants to be determined. To find ϕ_1 and ϕ_2, Eq. (4.6-2) is first differentiated and substituted for in Eq. (4.2-4). In addition, Eq. (4.6-2) is substituted directly for in Eq. (4.2-4) and Eq. (4.6-1). The integral term in Eq. (4.6-1) is evaluated analytically. This yields a single equation in the two unknowns ϕ_1 and ϕ_2. To obtain ϕ_1 and ϕ_2, this equation is evaluated for the full time step Δt, and for the half time step $\Delta t/2$, to give two linear equations in the two unknowns. Solution of this equation set yields ϕ_1 and ϕ_2, and thus an estimate of $\phi(t + \Delta t)$.

To control the precision of this approximate solution method, an internal time-step control algorithm has been implemented. In addition to the full time-step solution, a half time step solution is also obtained (i.e. evaluation of Eq. (4.2-4) at $\Delta t/2$ and $\Delta t/4$). The half-step solution is extrapolated to the end of the time step and compared with the full step solution. If the fractional difference is within a user-specified tolerance, the full-step solution is accepted. If not, the internal time step is halved and the entire solution process is repeated. This procedure is carried out until the solution is advanced to the end of the main time step. The half/full-step fission power amplitude convergence precision is entered as **EPSPOW**.

On each main time step, the point kinetics solution is obtained twice. At the beginning of the time step, the net reactivity is linearly extrapolated in time to the end of the time step, and a solution for ϕ is obtained with this extrapolated value. This solution is then used to drive the energy equation solutions in the SAS4A/SASSYS-1 models through the time step. When all channels have been advanced to the end of the main time step, the reactivities are calculated and the net reactivity is used to solve once more for ϕ. This second solution is taken as the final value for the time step, and solutions for the delayed-neutron precursor and decay-heat equations are based on this value.

For long, slow transients, an oscillation or instability in the solution for reactivity and power has been observed when time steps larger than 0.25 seconds are used. Modifications to the code have been made to eliminate this instability or oscillation. With the modified code, time step sizes of 1 second can be used in the early part of the transient, when flows and temperatures are changing rapidly, and 5 second step sizes can be used after the first few hundred seconds of the transient. The oscillations or instabilities come about because the reactivity used to compute the power level for a time step is an extrapolated reactivity, based on the last two previous computed steps. The extrapolated reactivity is used to compute a power level that is then used to compute fuel pin temperatures. After the temperatures are computed, the reactivity is computed. The computed reactivity may not be the same as the extrapolated reactivity. If the extrapolated reactivity is high, it will lead to higher fuel temperatures, which will lead to a lower computed reactivity. For the next step the extrapolated reactivity will be low, and the computed reactivity will be high. For large time steps, this leads to oscillations. The fuel expansion and Doppler reactivity oscillate up-down-up-down from step to step.

Adding a correction term to the extrapolated reactivity, k^n_{ext}, for time step n eliminated the oscillations. First an error, e_{n}, is calculated as the difference between the computed reactivity, k^n, and the extrapolated reactivity:

$$e_{\mathrm{n}} = k^n - k^n_{\mathrm{ext}} \tag{4.6-3}$$

Then the reactivity k' actually used for step n is computed as

$$k' = k^n_{\mathrm{ext}} + |e_{\mathrm{n}-1}|\, \mathrm{sign}\,(e_{\mathrm{n}-2}) \tag{4.6-4}$$

The correction term tends to cancel out the oscillations. The application of this correction is controlled by input variable **JREEXT**.

4.6.2 Specified Power History

The point kinetics and decay-heat models may be overridden through the specification on input of a user-supplied power history. This option is intended to be used for analyses in which the power history is known, such as those for in-pile experiments. It is also possible to specify an external, driving reactivity that is summed with the internally calculated feedback reactivity. This option may employ either a user-specified subroutine function named PREA, or a user-specified table of input power (or reactivity) values for input problem time values. The choice of specifying power or external reactivity is controlled with input variable **IPOWER**, and the choice of an input function or table is set by **NPREAT**, which gives the number of entries in the power (or reactivity) table and defaults to zero, which indicates usage of the input function PREA. Should a PREA not be supplied for this case, steady-state power or reactivity conditions are assumed.

For **NPREAT** > 0, input values for the power (or reactivity) and problem times at which they apply are entered in arrays **PREATB** and **PREATM**.

When the power vs. time option is invoked (**IPOWER** = 1), additional power tables may be entered by setting **NPOWDK** to the total number of tables to be input (2, …, 5), and entering the additional power vs. time table in arrays **PRETB2** and **PRETM2**. (The first table is always entered in **PREATB** and **PREATM**). Input integer **IDKCRV** then specifies the input power curve to be used in a particular channel. Total reactor power reported in the output represents only channels assigned to the first input power curve and does not include power from channels assigned to input power curves 2-5. Note that this multiple input power table option is only available for **IPOWER** = 1 and **NPOWDK** > 1.

4.7 Code Organization and Data Flow

The subroutines in SAS4A/SASSYS-1 that are mainly related to the solution of the reactor kinetics, decay-heat, and reactivity feedback models are listed in Table 4.7.1. Most of these routines are called directly from the steady-state and transient driver routines, SSTHRM and TSTHRM. Flow diagrams for these routines are included in Section 2.2.

Table 4.7.1: Listing of Reactor Kinetics Subroutines

Subroutine	Description
SSPK	Initializes all steady-state point kinetics data including power, reactivity, delayed-neutron precursor concentrations, and decay heat sources.
POWING	Extrapolates power at the beginning of a main time step for external power vs. time option via call to PREA. Updates parabolic power history coefficients.
PREA	User-supplied external power or reactivity history.
PAR	Fits a parabola to three points.
POINEX	Extrapolates reactivity at the beginning of a main time step for reactivity vs. time option by parabolic extrapolation of calculated internal reactivity and direct evaluation of external reactivity via call to PREA.
TSPK	Given an extrapolated or calculated reactivity, drives the calculation of the reactor power. Resets beginning time-step values and calls PKSTEP.
PKSTEP	Performs the integration of the point kinetics equations across a main time step.
RHOEND	Updates end-of-time step values.
FEEDBK	Performs calculation of Doppler and coolant void reactivity feedback for a channel.

At steady state, the driver routine SSTHRM calls subroutine SSPK, which performs all required data initialization. In the transient calculation, the point kinetics equations are solved on each main time step. From driver

routine TSTHRM, subroutine POWING is called at the beginning of each main time step to provide an extrapolation in time of the reactor power for the power versus time option (IPOWER = 1). For the reactivity versus time option (IPOWER = 0), subroutine POINEX is called to provide an extrapolation of the reactivity, and subroutine TSPK is then called to compute the reactor power given the reactivity. Subroutine TSPK performs data initialization and calls subroutine PKSTEP, which integrates the point kinetics and decay-heat equations over the main time step. At the end of the main time step, subroutine FEEDBK is called from driver routine TSTHRM to compute the current reactivity feedback, and subroutine TSPK is called once more to compute the final estimate of the reactor power. Subroutine RHOEND is then called to update the coefficients used in the integration of the point kinetics equations.

4.8 Input/Output Description

Table 4.8.1 contains a listing of SAS4A/SASSYS-1 input locations that are relevant to the point kinetics and reactivity feedback models. Table 4.8.2 contains a listing of input locations that are relevant to the new decay heat model. A complete description of all SAS4A/SASSYS-1 input may be found in Section 2.7.

The new decay heat model is compatible with old input decks. However, to support the larger number of terms in the ANS standard as well as specifying multiple curves within a single decay heat region, new input locations have been specified. The new decay heat model still supports a total of 5 user-supplied decay heat curves, but also supports up to eight built-in standard curves. (Currently, only four are defined by the code.) A total of 10 decay heat regions can be defined using any combination of decay heat curves. User-supplied integer input for the decay heat model is contained in Block 1 (INPCOM). Floating-point input is contained in Block 12 (POWINA).

The simplest usage of the new decay heat model is one that takes advantage of the built-in standard curves. For example, a problem with a single decay heat region that can be characterized by the decay heat from U-235 fission requires only two input parameters: NDKGRP = 23 and DKANSI(1,1) = 1.0.

In addition to the decay heat input described below, the channel-dependent input parameter **IDKCRV** must be specified to indicate which decay heat *region* each channel belongs. If this value is zero, no decay heat will be calculated for that channel, and power from the predefined non-decay-heat region will be used. Similarly, the association of bypass channels with decay heat regions is defined by **IDKTYP**.

On each main time step, the following quantities are printed: a) total reactor power (normalized to unity at time zero), b) decay-heat power, c) integrated energy (full-power seconds), d) net reactivity, e) programmed reactivity, f) control rod drive expansion reactivity, g) Doppler reactivity, h) fuel and cladding axial expansion reactivity, i) coolant reactivity, j) fuel-motion reactivity, and k) cladding-motion reactivity. In addition, the coolant, fuel, cladding, axial expansion and Doppler reactivity for each channel are also printed at each main time step. This information is also printed following each major print in expanded form (more significant digits), along with the group-wise values of the delayed-neutron precursor concentrations and decay-heat powers.

Table 4.8.1: Point Kinetics and Reactivity Feedback Input Data

Symbol	Reference	Name	Block	Location	Suggested Value
-	-	IPOWER	1	8	0 or 1
-	4.11.1	IPOWOP	1	9	0 or 1
-	-	NPK	1	10	0
-	-	NDELAY	1	16	≥0, ≤6
-	-	NPREAT	1	18	≥0, ≤20
-	-	ICREXP	1	31	≥0, ≤3
-	-	IDBPWI	1	35	0 or 1
-	-	IRADEX	1	36	≥−7, ≤7

continues on next page

Table 4.8.1 – continued from previous page

Symbol	Reference	Name	Block	Location	Suggested Value
-	-	NFUELD	1	39	<0
-	-	NOREAC	1	41	≥0
-	-	NSRMTB	1	43	Not used.
-	-	NSUBTC	1	51	≥0
-	-	MTGRD	1	52	1 or 2
-	-	MTACLP	1	53	1 or 2
-	-	MTTLP	1	54	1 or 2
-	-	MODEEX	1	55	≥0, ≤3
-	-	JREEXT	1	56	0 or 1
-	-	IREACT	1	58	≥−2, ≤2
-	-	NSUBTR	1	59	≥0
-	-	NRRNGS	1	60	0,1, or 2
-	-	MTRRAC	1	61	1 or 2
-	-	MTRFAC	1	63	1 or 2
-	-	MTRFT	1	64	1 or 2
-	-	IROPT	1	65	0 or 1
-	-	JCRIND	1	66	>0
-	-	ICRDDB	1	71	0,1, or 2
-	-	ICRTMP	1	72-74	≥0
-	-	ICRNOD	1	75-77	-
-	-	NSEGCR	1	78	>0
-	-	MTCB	1	81	1 or 2
-	-	KDEBUG	1	82	0 or 1
-	-	KEBRS1	1	83	≥0
-	-	KEBRS2	1	84	≥0
-	-	IDBDKH	1	85	-
-	-	NULLD3	1	86	-
-	-	NOFDBK	1	89	≥0
-	-	IBOWTP	1	102	0 or 1
-	-	KCHUIS	3	1209-1242	Not used.
-	-	NRREAC	3	1292	≥0
-	-	ISLREA	3	1293-1300	≥0
-	-	LBYP	3	1301	≥0
-	-	LELBYP	3	1302-1309	≥0
-	-	EPSPOW	11	3	10^{-4}
-	-	ASCRAM	11	23	Not used.
-	-	PSCRAM	11	24	Not used.
-	-	GSCRAM	11	25	Not used.
-	4.11.1	POW	12	1	-
Λ	Eq. (4.2-4)	GENTIM	12	2	-
-	4.11.1	POWTOT	12	3	-
β_i	Eq. (4.3-1)	BETADN	12	4-9	-
λ_i	Eq. (4.3-1)	DECCON	12	10-15	-
-	-	OLDBDK	12	16-21	Not used.
-	-	OLDDKL	12	22-27	Not used.
-	-	OLDBDT	12	28	Not used.
-	-	PREATB	12	29	-
-	-	PREATM	12	49	-
-	4.11.1	FRPR	12	69	-

continues on next page

Table 4.8.1 – continued from previous page

Symbol	Reference	Name	Block	Location	Suggested Value
L_{cr}	Eq. (4.5-49)	CRDLEN	12	71	-
α_{cr}	Eq. (4.5-49)	CRDEXP	12	72	-
a_{cr}	Eq. (4.5-52)	ACRDEX	12	73	-
b_{cr}	Eq. (4.5-52)	BCRDEX	12	74	-
$M_{cr}C_{cr}$	Eq. (4.5-42)	CRDMC	12	75	-
$h_{cr}A_{cr}$	Eq. (4.5-42)	CRDHA	12	76	-
V_{ui}	Eq. (4.5-43)	UIVOL	12	77	-
C_{re}	Eq. (4.5-27)	RDEXPC	12	78	-
$\frac{XMC}{XAC}$	Eq. (4.5-27)	XMCXAC	12	79	-
-	-	SCRTAB	12	80	Not used.
-	-	SCRTME	12	90	Not used.
-	-	PRETB2	12	100-179	-
-	-	PRETM2	12	180-259	-
S_{GRMAX}	Eq. (4.5-31)	SLLMAX	12	408	-
-	-	PITCHG	12	409	-
-	-	PITCHA	12	410	-
-	-	PITCHT	12	411	-
-	-	RDEXCF	12	412	-
-	-	TLPRRC	12	413	-
-	-	BNDMM1	12	414	-
-	-	BNDMM2	12	415	-
-	-	TINSRT	12	416	-
-	-	REAINS	12	417	-
-	-	TLIMIT	12	418	-
-	-	DFLTCS	12	419	-
-	-	DFLTSS	12	420	-
-	-	ACLPRC	12	421	-
-	-	FCDTR1	12	422	-
-	-	FCDTR2	12	423	-
-	-	FCDTRF	12	424	-
-	-	DRCOLL	12	425	-
-	-	CRSAC	12	426	-
-	-	RR1TC	12	427	-
-	-	RR2TC	12	428	-
-	-	RODID	12	429	-
-	-	RODOD	12	430	-
-	-	SHRDLN	12	431-433	-
-	-	SHRDID	12	434-436	-
-	-	SHRDOD	12	437-439	-
-	-	RHOCRD	12	440	-
-	-	HTCPCR	12	441	-
-	-	CONDCR	12	442	-
-	-	VFCRD	12	443	-
-	-	HFILM	12	444	-
-	-	FLSHRD	12	445	-
-	-	AREACR	12	446	-
-	-	FLOEXP	12	447	-
-	-	ACLPEL	12	448	-
-	-	TLPEL	12	449	-

continues on next page

Table 4.8.1 – continued from previous page

Symbol	Reference	Name	Block	Location	Suggested Value
-	-	PTCHRA	12	450	-
-	-	PTCHRT	12	451	-
-	-	RCBARR	12	452	-
-	-	FCDTCB	12	453	-
-	-	CB2TC	12	454	-
-	-	YKNF	12	455	-
-	-	YKHF	12	456	-
-	-	YKNNA	12	457	-
-	-	YKNSS	12	458	-
-	-	YRCUR	12	459	-
-	-	YRCLR	12	460	-
-	-	YRCRR	12	461	-
-	-	YRCCR	12	462	-
-	-	YRCGP	12	463	-
-	-	YRCDOP	12	464	-
-	-	YDELT0	12	465	-
-	-	YABOW	12	466	-
-	-	YBBOW	12	467	-
-	-	FCR	12	468	-
-	-	YTCUT	12	469	-
ε_{ex}	-	EXPCOF	13	1263	-
-	-	IAXEXP	51	181	0, 1, or 2
-	-	IDKCRV	51	203	≥0, ≤5
-	-	ICHUIS	51	362	0 or 1
-	-	IPOWRZ	51	364	0 or 1
-	-	IREACZ	51	365	0 or 1
γ_s	Eq. (3.3-23)	GAMSS	62	2	-
γ_c	Eq. (3.3-23)	GAMTNC	62	4	-
γ_e	Eq. (3.3-23)	GAMTNE	62	5	-
$\overline{P}$	4.11.1	PSHAPE	62	6-29	-
P_r	-	PSHAPR	62	30-44	-
α_D	Eq. (4.5-2)	ADOP	62	62	-
α_D	Eq. (4.5-2)	BDOP	62	63	-
-	-	WDOPA	62	64-111	-
$(\rho_c)_{jI}$	Eq. (4.5-25)	VOIDRA	62	112-159	-
R_e	-	CLADRA	62	160-207	-
R_f	Eq. (4.5-21)	FUELRA	62	208-255	-
-	4.11.1	PRSHAP	62	256	-
-	-	PSHPTP	62	257-261	-
-	-	PSHPBT	62	262-266	-
-	-	XRFSHP	62	268	-
-	-	XRNSHP	62	269	-
-	-	XRSSHP	62	270	-
-	-	PSHAPC	62	271-318	-
-	-	PSHAPB	62	319-366	-
α_f	Eq. (4.5-4)	FUELEX	63	73	-
α_e	Eq. (4.5-5)	CLADEX	63	74	-
Y_f	Eq. (4.5-8)	YFUEL	63	75	-
Y_e	Eq. (4.5-8)	YCLAD	63	76	-

continues on next page

Table 4.8.1 – continued from previous page

Symbol	Reference	Name	Block	Location	Suggested Value
ε_{ex}	Eq. (4.5-24)	EXPCFF	63	79	-

Table 4.8.2: Decay Heat Input Data

Symbol	Reference	Name	Block	Location	Comments
-	Eq. (4.4-19)	NDKGRP	1	17	max = 24
-		NPOWDK	1	45	max = 5
-		NPDKST	1	46	max = 8
β_n	Eq. (4.4-4)	BETADK *or* DKBET2	12	260-289 511-630	deprecated, use DKBET2
λ_n	Eq. (4.4-1)	DKLAM *or* DKLAM2	12	290-319 631-750	deprecated, use DKLAM2
β_d	Eq. (4.4-4)	BETAHT	12	320-324	-
x_i	Eq. (4.4-12)	DKFRAC	12	751-800	-
x_i	Eq. (4.4-12)	DKANSI	12	801-880	-
P_t^k	Eq. (4.4-19)	POWLVL *and* PWLVL2	12	325-364 881-920	PWLVL2 is an extension of POWLVL
τ_k	Eq. (4.4-19)	POWTIM *and* PWTIM2	12	365-404 921-960	PWTIM2 is an extension of POWTIM
Q_t	Eq. (4.4-2)	QETOT	12	961-968	default = 200 MeV

4.9 References

4-1. F. E. Dunn, et al., Unpublished information, Argonne National Laboratory, 1975.

4-2. A. F. Henry, "The Application of Reactor Kinetics to the Analysis of Experiment," *Nuclear Science and Engineering*, vol. 3, p. 52, 1958.

4-3. G. I. Bell and S. Glasstone, *Nuclear Reactor Theory*, Van Nostrand Reinhold Company, New York, N.Y., 1970.

4-4. *American National Standard Decay Heat Power in Light Water Reactors*, ANSI/ANS-5.1-2005, American Nuclear Society, 2005.

4-5. P. R. Huebotter, Personal Communication, June 3, 1981.

4-6. G. A. McLennan, Unpublished information, Argonne National Laboratory, 1978.

4-7. K. L. Basehore and N. E. Todreas, "SUPERENERGY-2: A Multiassembly, Steady-State Computer Code for LMFBR Core Thermal-Hydraulics Analysis," Pacific Northwest Laboratory, Unpublished information, August, 1980.

4-8. L. K. Chang and D. Mohr, "The Effect of Primary Pump Coastdown Characteristics on Unprotected Loss-of-Flow Transients in EBR-II," Second Specialists' Meeting on Decay Heat Removal and Natural Circulation in LMFBRs, Brookhaven National Laboratory, April 17-19, 1985.

4-9. D. Mohr and L. K. Chang, "Perturbations of Reactor Flow and Inlet Temperature for EBR-II Reactivity-Feedback Validation," *Proceedings of the International Topical Meeting on Fast Reactor Safety*, Knoxville, Tenn., April 21-25, 1985.

4-10. M. L. Carboneau, et al., Unpublished information, Argonne National Laboratory, 1980.

4-11. A. M. White, Personal Communication, February 26, 1993.

4-12. J. P. Herzog, Personal Communication, October 20, 1993.

4-13. J. J. Kaganove, et al., "EBR-II Reactor Operation Simulator: Volume 1. A description of EROS," ANL-7981, Argonne National Laboratory, August, 1974.

4-14. J. J. Kaganove, "Numerical Solution of the One-Group Space-Independent Reactor Kinetics Equations for Neutron Density Given the Excess Reactivity," ANL-6132, Argonne National Laboratory, February, 1960.

4-15. E. L. Fuller, "The Point Kinetics Algorithm for FX2," ANL-7910, Argonne National Laboratory, pp. 503-508, 1972.

4.10 Nomenclature

Symbol	Description	Units
A	Coefficient in EBR-II bowing reactivity.	\$
A_{cr}	Control rod driveline heat transfer area.	m^2
A_e	Nominal cross-sectional area of cladding.	m^2
A_f	Nominal cross-sectional area of fuel.	m^2
a	Above-core load pad elevation.	m
a_{cr}	Coefficient in control rod driveline reactivity feedback.	\$/m
B	Coefficient in EBR-II bowing reactivity.	\$
b_{cr}	Coefficient in control rod driveline reactivity feedback.	$/m^2
C_{cr}	Control rod driveline specific heat.	J/kg-K
C_i	Delayed neutron precursor group i population.	
C_{rc}	Coefficient in simple radial core expansion reactivity feedback.	\$/K
D	Hex can flat-to-flat dimension.	m
D_z	Nominal axial node height in simple axial expansion reactivity feedback model.	m
E	Modulus of elasticity.	N/m^2
F_{cr}	Temperature multiplier in EBR-II control rod bank expansion model.	
F_{LowBU}	Fraction of fuel elements with burnup less than 2.9% in EBR-II.	
f_e	Cladding mesh height expansion factor.	
f_f	Fuel mesh height expansion factor.	
f_i	Channel flag in simple radial expansion reactivity model.	
H	Fuel height in EBR-II.	m
h_{cr}	Coolant to control rod driveline heat transfer coefficient.	W/m^2-k
H_n	Normalized decay heat energy fraction for group n.	
I	Moment of inertia of beam cross-sectional area.	m^4
k	Reactor effective multiplication constant.	
δk	Reactivity, or change in the effective reactor multiplication constant.	

continues on next page

Table 4.10.1 – continued from previous page

Symbol	Description	Units
δk_{cl}	Cladding relocation reactivity.	
δk_{cr}	Control rod drive expansion reactivity.	
δk_{cs}	Control system reactivity.	
δk_{D}	Fuel Doppler reactivity.	
δk_{d}	Fuel and cladding axial expansion reactivity.	
δk_{fu}	Fuel relocation reactivity.	
δk_{Na}	Coolant density or voiding reactivity.	
δk_{p}	User-programmed reactivity.	
δk_{re}	Core radial expansion reactivity.	
L	Elevation of top load pad.	m
L_{cr}	Control rod driveline length.	m
L_{k}	Reactor vessel wall length represented by compressible volume k or element k.	m
M	Bending moment.	N-m
M_{cr}	Control rod driveline mass.	kg
M_{GR}	Applied moment at grid plate.	N-m
M_1	Thermally induced bending moment in the core region.	N-m
M_2	Thermally induced bending moment above the core region.	N-m
m_{ij}	Fuel or cladding mass at node j in channel i in fuel and cladding relocation reactivity.	kg
N	Fuel, sodium, or steel atom density in EBR-II reactivity model.	atm/m^3
N_{c}	Number of channels.	
N_{s}	Number of subassemblies per channel.	
P	Radial force at the above core load pad.	N
P_{t}^k	Normalized power in interval k of irradiation history for decay heat precursor initial condition.	
Q	Space and time dependent nodal power.	W
Q_{t}	Total recoverable energy per fission	MeV
R_{e}	Cladding nodal reactivity worth in simple axial expansion reactivity model.	kg^{-1}
R_{f}	Fuel nodal reactivity worth in simple axial expansion reactivity model.	kg^{-1}
R_{i}	Radial fuel reactivity worth shape factor in EBR-II reactivity model.	
R_1	Minimum core radius at above-core load pad in detailed radial expansion model.	m
R_2	Minimum core radius at top load pad in detailed radial expansion model.	m
R_3	Maximum core radius at restraint ring in detailed radial expansion.	m
$\overrightarrow{r}$	Reactor spatial position vector.	
S	Space-dependent steady-state nodal power.	W
S_{GR}	Subassembly slope with respect to vertical at the grid plate.	m/m
S_{GRMAX}	Maximum subassembly slope with respect to vertical at the grid plate.	m/m
T	Temperature.	K
$\overline{T}$	Average temperature.	K
ΔT	Hexcan flat-to-flat temperature difference.	K
T_{by}	Bypass region temperature.	K
T_{ch}	Reflector steel temperature.	K
T_{cr}	Control rod driveline temperature.	K
T_{cv1}	Inlet plenum coolant temperature.	K
T_{cv2}	Outlet plenum coolant temperature.	K

continues on next page

Table 4.10.1 – continued from previous page

Symbol	Description	Units
T_{e}	Cladding temperature.	K
T_{f}	Fuel temperature.	K
$\overline{T}_{\mathrm{f}}$	Average fuel temperature.	K
ΔT_f	Fuel temperature change.	K
$\Delta \overline{T}_{\mathrm{favg}}$	Average fuel temperature change.	K
$\overline{T}_{\mathrm{floc}}$	Average fuel temperature for a channel.	K
T_{i}	Plenum coolant temperature.	K
T_{in}	Coolant inlet temperature.	K
ΔT_{in}	Coolant inlet temperature change.	K
T_{j}	Reflector coolant temperature.	K
$\overline{T}_{\mathrm{k}}$	Average vessel wall temperature in k-th compressible volume or element.	K
$\overline{T}_{\mathrm{lr}}$	Average coolant temperature in lower reflector.	K
T_{mm}	Mixed mean coolant core outlet temperature.	K
$\overline{T}_{\mathrm{Na}}$	Average coolant temperature.	K
$\overline{T}_{\mathrm{rr}}$	Average radial reflector coolant temperature.	K
T_{SLP}	Space and time dependent structure temperature.	K
$\overline{T}_{\mathrm{SLP}}$	Space-average, time dependent structure temperature.	K
$\Delta \overline{T}_{\mathrm{SLP}}$	Change in space-averaged structure temperature.	K
ΔT_{ss}	Cladding temperature change.	K
$\overline{T}_{\mathrm{ss}}$	Average cladding temperature.	K
T_{ui}	Coolant temperature in the upper internal structure region.	K
$\overline{T}_{\mathrm{ur}}$	Average coolant temperature in the upper reflector.	K
t	Time.	s
Δt	Change in time.	s
V_{by}	Bypass region volume.	m^3
$V_{\mathrm{c}}h$	Channel steel volume.	m^3
V_{cv1}	Inlet plenum volume.	m^3
V_{cv2}	Outlet plenum volume.	m^3
V_{f}	Fuel volume.	m^3
V_{GR}	Radial reaction at the grid plate.	
V_{i}	Plenum coolant volume.	m^3
V_{i}	Channel volume.	m^3
V_{j}	Reflector coolant volume.	m^3
V_{k}	Nodal volume at k in channel.	m^3
V_{Na}	Sodium volume in fuel/cladding gap.	m^3
V_{ss}	Cladding volume.	m^3
V_{ui}	Coolant volume in upper internal structure.	m^3
w_{c}	Core outlet flow rate.	kg/s
w_{k}	Core volume weighting for decay heat curve k.	
XAC	Distance from subassembly nozzle support to above core load pad.	m
XMC	Distance from subassembly nozzle support to core midplane.	m
x	Axial elevation in detailed radial core expansion model.	m
x_1	Elevation of lower axial blanket/lower reflector interface.	m
x_{i}	Fraction of total fission power from isotope i.	
Y	Fraction of full power temperature rise.	
Y_{e}	Cladding Young's modulus.	
Y_{f}	Fuel Young's modulus.	

continues on next page

Table 4.10.1 – continued from previous page

Symbol	Description	Units
Y_{ss}	Cladding Young's modulus.	
y	Radial displacement with respect to the core radius at the grid plate.	m
Z_k	Axial fuel reactivity worth shape.	
Δz_{cr}	Control rod driveline expansion.	m
z_0	Unexpanded axial mesh elevation.	m
Δz_n	Net control rod movement.	m
z_{ne}	Expanded cladding mesh elevation.	m
z_{nf}	Expanded fuel mesh elevation.	m
Δz_v	Reactor vessel expansion.	m
α	Subassembly hex can thermal expansion coefficient.	1/K
α_{cr}	Control rod driveline thermal expansion coefficients.	1/K
α_D	Local fuel Doppler coefficient.	
α_e	Cladding thermal expansion coefficient.	1/K
α_f	Fuel thermal expansion coefficient.	1/K
α_{jI}	Average coolant void fraction in axial node j of channel I.	
α_k	Reactor vessel thermal expansion coefficient.	1/K
α_{Lf}	Fuel thermal expansion coefficient.	1/K
α_{Lss}	Cladding thermal expansion coefficient.	1/K
α_{VNa}	Sodium volumetric thermal expansion coefficient.	1/K
α_n	Contribution of group n to immediate decay heat from a single fission event	MeV
β	Total effective delayed-neutron fraction	
β_n	Effective decay-heat power fraction for group n.	
β_i	Effective delayed-neutron precursor group i fraction.	
δ_e	Cladding axial expansion.	m
δ_f	Fuel axial expansion.	m
ϵ_d	Sodium density multiplier.	
ϵ_e	Cladding fractional thermal expansion.	
ϵ_{ex}	Effective axial expansion multiplier.	
ϵ_f	Fuel fractional thermal expansion.	
ζ	EBR-II reactivity parameter.	$
ζ_{bw}	EBR-II bowing reactivity parameter.	$
ζ_c	EBR-II coolant reactivity parameter.	$
ζ_{cr}	EBR-II control rod driveline reactivity parameter.	$
ζ_{fa}	EBR-II axial fuel expansion reactivity parameter.	$
ζ_{fr}	EBR-II radial fuel expansion reactivity parameter.	$
ζ_{lr}	EBR-II lower reflector reactivity parameter.	$
ζ_{rr}	EBR-II radial reflector reactivity parameter.	$
ζ_{ssa}	EBR-II axial cladding expansion reactivity parameter.	$
ζ_{ssr}	EBR-II radial cladding expansion reactivity parameter.	$
ζ_{ur}	EBR-II upper reflector reactivity parameter.	$
Λ	Prompt neutron lifetime.	s
λ_n	Decay heat group effective decay constant for group n.	1/s
λ_i	Delayed neutron precursor group i decay constant.	1/s
ρ_c	Local coolant void reactivity worth.	g^{-1}
$\Delta\rho_e$	Cladding axial expansion reactivity.	
$\Delta\rho_f$	Fuel axial expansion reactivity.	
ρ_u	Sodium density in outlet plenum.	kg/m^3
ϕ	Normalized fission power amplitude.	

continues on next page

Table 4.10.1 – continued from previous page

Symbol	Description	Units
ϕ_1	First order coefficient in normalized fission power amplitude expansion.	s^{-1}
ϕ_2	Second order coefficient in normalized fission power amplitude expansion.	s^{-2}
ψ_f	Normalized fission power.	
ψ_h	Normalized decay heat power.	
ψ_t	Normalized total power.	

4.11 Appendices

4.11.1 Appendix 4.1: Steady-State Power Normalization in SAS4A/SASSYS-1

The power in axial fuel pin segment I of channel J is given by the expression

$$\text{PSHAPE}\,(I,J)\ \times\ \text{QMULT}\times\ \text{POW}$$

Here POW is the maximum power of any axial fuel pin segment. QMULT is a multiplier that is equal to one in steady state.

PSHAPE (I, J) is the ratio of the power of segment (I, J) to the maximum power of any segment. It is obvious that PSHAPE takes values between zero and one.

SAS4A/SASSYS-1 will renormalize the input values of PSHAPE and PRSHAP. This is done is subroutine PNORM. The user needs only supply un-normalized data for these arrays.

PNORM will also compute POW from POWTOT or POWTOT from POW according to the value of IPOWOP.

Input:

IPOWOP = 0	Calculate steady-state power in peak axial segment from total reactor power.
IPOWOP = 1	Calculate steady-state total reactor power from the peak axial fuel pin segment.
POWTOT	Total reactor power in steady state.
POW	Steady-state power in the peak axial pin segment.
FRPR	Fraction of total reactor power represented by sum of all SAS4A/SASSYS-1 channels.
NPIN(ICH)	Number of fuel pins in a subassembly of channel ICH.
NSUBAS(ICH)	Number of subassemblies in channel ICH.
MZ(ICH)	Number of axial nodes in channel ICH.
PRSHAP(ICH)	The relative power per *subassembly* in channel ICH. PRSHAP will be normalized by PNORM.
PSHAPE(I,ICH)	The relative power per axial *pin* segment of axial pin segment I and channel ICH. I = 1, … MZ(ICH). PSHAPE will be normalized by the PNORM routine.

Method:

The values of PRSHAP get renormalized as RELCHA for all channels:

$$\text{RELCHA}\,(\text{ICH}) = \text{PRSHAP}\,(\text{ICH})\ \times \frac{\sum_{\text{I}} \text{NSUBAS}\,(I)}{\sum_{\text{I}} \text{PRSHAP}\,(I) \times \text{NSUBAS}\,(I)}$$

If we multiply the above equation by NSUBAS (ICH) and then sum over all channels ICH, we can show that

$$\sum_{\text{ICH}} \text{RELCHA (ICH)} \times \text{NSUBAS (ICH)} = \sum_{\text{ICH}} \text{NSUBAS (ICH)}$$

Thus, RELCHA is properly normalized.

The values of PSHAPE get normalized as RELSHP for each channel:

$$\text{RELSHP}\,(IZ, ICH) = \frac{\text{PSHAPE}\,(IZ, ICH)}{\sum_{\text{I}} \text{PSHAPE}\,(I, ICH)}$$

If we sum over all axial nodes IZ, we can show that

$$\sum_{\text{IZ}} \text{RELSHP}\,(IZ, ICH) = 1$$

Thus, RELSHP is properly normalized.

The power of axial pin segment (I, J) is

$$\text{RELSHP}\,(I, J) \times \frac{\text{RELCHA}\,(J)}{\text{NPIN}\,(J)} \times \frac{\text{FRPR} \times \text{POWTOT}}{\sum_{\text{ICH}} \text{NSUBAS (ICH)}}$$

From the definition of POW we get the equation

$$\text{POW} = \left[\text{RELSHP}\,(I, J) \times \frac{\text{RELCHA}\,(J)}{\text{NPIN}\,(J)}\right] \times \frac{\text{FRPR} \times \text{POWTOT}}{\sum_{\text{ICH}} \text{NSUBAS (ICH)}}$$

and the unknown POWTOT or POW can be found. Given I_{max} and J_{max} to be the indices of the peak power pin segment, we then redefine PSHAPE (I, J) as

$$\text{PSHAPE}\,(I, J) = \frac{\text{RELSHP}\,(I, J) \times \text{RELCHA}\,(J)}{\text{NPIN}\,(J)} \times \frac{\text{NPIN}\,(J_{\text{max}})}{\text{RELSHP}\,(I_{\text{max}}, J_{\text{max}}) \times \text{RELCHA}\,(J_{\text{max}})}$$

This is the desired form of PSHAPE that is used by the subroutine SHAPE.

CHAPTER

FIVE

PRIMARY AND INTERMEDIATE LOOP THERMAL HYDRAULICS MODULE

5.1 Introduction

The PRIMAR-4 module computes coolant pressures, flow rates, and temperatures in the primary and intermediate heat transport loops. This module is designed for analysis of a wide range of transients, from fast unprotected LOF or TOP cases to slow operational transients or natural circulation shut-down heat-removal cases. Also, an arbitrary arrangement of components in either a loop-type or a pool-type system can be treated. Semi-implicit and fully implicit numerical schemes have been developed to handle the full range of transients efficiently. This chapter describes the physical models and the numerical algorithms used in PRIMAR-4. As indicated in Figure 5.1.1, PRIMAR-4 couples with the driver and blanket subassembly coolant dynamics calculations in the pre-voiding thermal hydraulics module TSCL0, and the boiling module TSBOIL. Also, PRIMAR-4 couples with the detailed steam generator model described in Chapter 7, and with the control system described in Chapter 6.

The PRIMAR-4 module contains both a simple PRIMAR-1 type option and the more detailed PRIMAR-4 treatment. The PRIMAR-1 option supplies only the minimum quantities of information required to drive the subassembly coolant dynamics calculations. It is often used when the more detailed treatment is not required. This minimum information consists of the inlet and outlet plenum pressures, the subassembly inlet temperatures, and the outlet plenum temperature in the event of flow reversal. All of these quantities are computed as functions of time by PRIMAR-1 from user-supplied information independently of what is happening in the core or in the rest of the primary loop.

The detailed multi-loop PRIMAR-4 treatment models heat transfer and coolant flow in the inlet and outlet coolant plenums, the pipes, pumps, and valves in the primary and intermediate loops, the intermediate heat exchangers, the steam generators, and the pool in a pool system. It utilizes a modular approach. The user specifies the properties of various components and arranges them in an arbitrary manner. Each type of component is treated in a separate section of the code. Components can be added or existing component treatments can be modified without impacting the rest of the code.

The detailed PRIMAR-4 treatment includes both a steady-state initialization sub-module and a transient sub-module. The main parts of the transient sub-module are a hydraulic section in which liquid coolant flow rates and pressures are calculated, a liquid temperature section, and a gas section that calculates gas flow between cover gasses. For a given time step, the liquid flow rates and pressures are calculated first, taking into account pressure and flow rate changes during the time step, but ignoring temperature changes and gas flow changes during the time step. This not only simplifies the calculations, but it makes explicit use of the fact that the incompressible single-phase liquid flow rates are more sensitive to pressure changes than they are to temperature and gas flow changes. Then, with the flow rates known, the liquid temperatures are calculated for the time step. Finally, with the liquid pressures, flow rates, and temperatures known, the gas flow rates and pressures are calculated, and the liquid pressures are adjusted to account for changes in the cover-gas pressures.

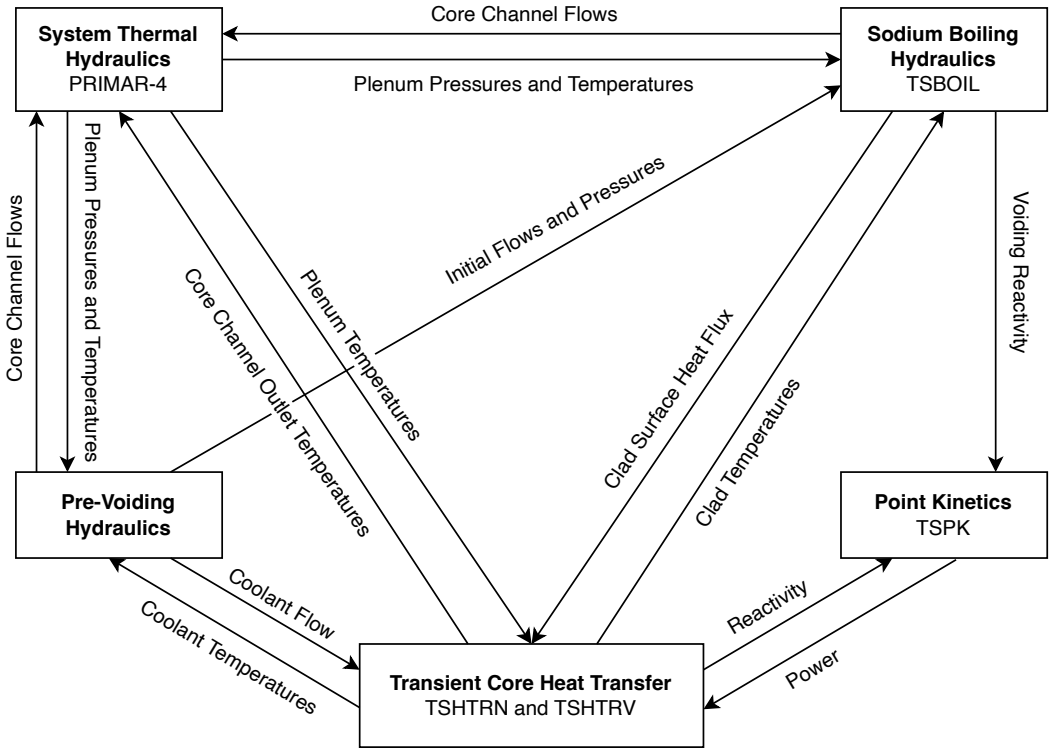


Figure 5.1.1: Interaction between PRIMAR-4 and Other Modules.

5.2 Hydraulic Calculations

5.2.1 Compressible Volumes and Liquid Segments

The primary and intermediate loop thermal hydraulics calculations use a generalized geometry, as indicated in Figure 5.2.1. A number of compressible volumes are connected by liquid or gas segments, and each liquid segment can contain one or more elements. The treatment allows compressible volumes and segments to be connected in an arbitrary manner. Table 5.2.1 lists the types of compressible volumes used in PRIMAR-4. Compressible volumes are characterized by pressure, volume, mass, and temperature. They can accumulate liquid or gas by compressing the cover gas or the liquid, and it is the pressure in the compressible volumes that drives the flows through the liquid and gas segments. Table 5.2.2 lists the types of elements that can make up a liquid segment. Liquid flow elements are characterized by incompressible single-phase flow, with the possible exception of the core element. The core subassemblies, which are treated by the coolant dynamics modules, are a special case and are discussed in Section 5.2.2.

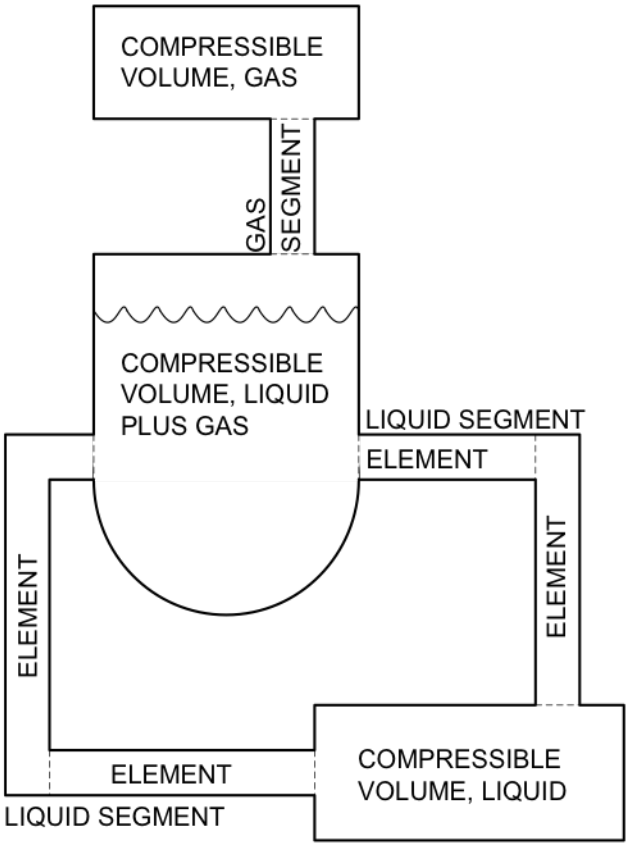


Figure 5.2.1: PRIMAR-4 Generalized Geometry.

Table 5.2.1: Compressible Volume Types Used in PRIMAR-4

Type Number	Description
1	Inlet plenum
2	Compressible liquid volume, no cover gas
3	Closed outlet plenum, no cover gas
4	Almost incompressible liquid junction, no cover gas
5	Pipe rupture source
6	Pipe rupture sink, guard vessel
7	Outlet plenum with cover gas
8	Pool
9	Pump bowl and cover gas
10	Expansion tank with cover gas
11	Compressible gas volume, no liquid

Table 5.2.2: Liquid Flow Element Types Used in PRIMAR-4

Type Number	Description
1	Core subassemblies, SAS channels
2	Core bypass assemblies
3	Pipe
4	Check valve
5	Pump
6	IHX, shell side
7	IHX, tube side
8	Steam generator, sodium side
9	DRACS heat exchanger, tube side
10	DRACS heat exchanger, shell side
11	Valve
12	Air dump heat exchanger, sodium side
13	Annular element
15	Annular pump

The hydraulic equations for the primary and intermediate heat-transport loops are solved by a semi-implicit or fully implicit time differencing scheme in which the pressures and flows for all connected compressible volumes and segments are solved for simultaneously. By linearizing the equations for each time step, a semi-implicit or fully implicit solution can be obtained without resorting to iteration techniques. Linearized semi-implicit or fully implicit methods are most useful for long transients in which temperatures and flows change slowly, since in such cases accurate results can be obtained with large time steps so long as the step sizes are small enough that changes during a step are small.

Three equations are used in calculating the pressures in the compressible volumes and the flow rates in the connecting liquid segments. They are the momentum equation for incompressible single-phase flow in a segment, an expression for the average flow rate in a segment during a time step, and an expression for the change in pressure in a compressible volume as a result of flow into it and out from it during a time step. Each of these equations is taken up in turn.

5.2.1.1 Momentum Equation

The momentum equation for a single-phase incompressible liquid is taken as

$$\frac{1}{A}\frac{\partial \mathrm{w}}{\partial \mathrm{t}}+\frac{1}{A^2}\frac{\partial}{\partial \mathrm{z}}\left(\frac{w^2}{\rho}\right)+\frac{\partial \mathrm{p}}{\partial \mathrm{z}}+\left(\frac{\partial \mathrm{p}}{\partial \mathrm{z}}\right)_{\mathrm{loss}}=0 \tag{5.2-1}$$

where

w = the mass flow rate

A = the flow area

ρ = the density of the liquid

$\frac{\partial \mathrm{p}}{\partial \mathrm{z}}$ = the pressure gradient driving the flow

$\left(\frac{\partial \mathrm{p}}{\partial \mathrm{z}}\right)_{\mathrm{loss}}$ = the pressure drop from all of the loss terms

When Eq. (5.2-1) is integrated over a segment containing several elements, it can be written as the basic equation for the flow in segment i:

$$\sum_{\mathrm{k}}\frac{L_{\mathrm{k}}}{A_{\mathrm{k}}}\frac{\mathrm{dw}\left(i\right)}{\mathrm{dt}}=p_{\mathrm{in}}\left(i\right)-p_{\mathrm{out}}\left(i\right)-\Delta p_{\mathrm{fr}}\left(i\right)-\Delta p_{\mathrm{w2}}\left(i\right)-\Delta p_{K_{in}}\left(i\right)-\Delta p_{\mathrm{v}}\left(i\right)-\Delta p_{\mathrm{gr}}\left(i\right)+\Delta p_{\mathrm{p}}\left(i\right) \tag{5.2-2}$$

Here, $\mathrm{dw}\left(i\right)/\mathrm{dt}$ is the time rate of change of the mass flow rate through segment i. The summation is over the elements in segment i, and L_{k} and A_{k} are the length and flow area of element k. The term $p_{\mathrm{in}}\left(i\right)$ is the pressure at the inlet to segment i, which is the pressure in the compressible volume at the inlet of segment i. The term $p_{\mathrm{out}}\left(i\right)$ is the pressure at the outlet of segment i, or in the compressible volume at the outlet of segment i. The term $\Delta p_{\mathrm{fr}}\left(i\right)$ is the frictional pressure change for the segment i, and with the minus sign in Eq. (5.2-2), it is a loss term. However, this frictional loss term for the segment is actually the sum of similar loss terms for each element in the segment. The same things can be said about the remaining terms in Eq. (5.2-2). Together with their respective signs, $\Delta p_{\mathrm{w2}}\left(i\right)$ is any orifice or bend pressure drop proportional to the square of the mass flow rate, $\Delta p_{\mathrm{v}}\left(i\right)$ is any valve pressure drop, $\Delta p_{\mathrm{gr}}\left(i\right)$ is the gravity-head pressure drop, and $\Delta p_{\mathrm{p}}\left(i\right)$ is the pump-head pressure increase from all of the pumps in segment i.

In Eq. (5.2-2), $\Delta p_{K_{in}}\left(i\right)$ is the minor pressure loss that occurs at the inlet of the segment. This term is defined as

$$\Delta p_{K_{in}}\left(i\right)=K_{in}\left(Re_{in}\right)\frac{w\left|w\right|}{2\rho A_{in}^2} \tag{5.2-3}$$

where $K_{out}\left(Re_{out}\right)$ is the anisotropic Re-dependent loss coefficient at the inlet of segment i, Re_{in} is the Re number at the inlet of segment i and A_{in} is the area at the inlet of segment i. Anisotropic Re-dependent loss coefficients at segment inlets are defined using **SegLossCoefTableID** and additional information can be found in Section 5.3.3.2.

Eq. (5.2-2) has the form:

$$\sum_{\mathrm{k}}\frac{L_{\mathrm{k}}}{A_{\mathrm{k}}}\frac{\mathrm{dw}\left(i\right)}{\mathrm{dt}}=f\left(w,t\right) \tag{5.2-4}$$

which can be written in finite difference form

$$\sum_{\mathrm{k}}\frac{L_{\mathrm{k}}}{A_{\mathrm{k}}}\frac{\Delta w\left(i\right)}{\Delta t}=\theta_1\left(i\right)f_{\mathrm{i}}\left(w,t\right)+\theta_2\left(i\right)f_{\mathrm{i}}(w+\Delta w,t+\Delta t) \tag{5.2-5}$$

where $\theta_1 + \theta_2 = 1$. The parameters θ_1 and θ_2 determine the degree of implicitness of the solution. For a fully explicit solution, $\theta_1 = 1$ and $\theta_2 = 0$. For a fully implicit solution $\theta_1 = 0$ and $\theta_2 = 1$. The degree of implicitness is discussed in Section 5.2.4 and in Section 3.19.1 in Chapter 3.

The linearization consists in making the approximation that

$$f\left(w + \Delta w, t + \Delta t\right) = f\left(w, t\right) + \Delta t \frac{\partial \mathrm{f}}{\partial \mathrm{t}} + \Delta w \frac{\partial \mathrm{f}}{\partial \mathrm{w}} \tag{5.2-6}$$

so the flow equation becomes

$$\sum_{\mathrm{k}} \frac{L_{\mathrm{k}}}{A_{\mathrm{k}}} \Delta w\left(i\right) = \Delta t \left\{ f\left(w, t\right) + \theta_2\left(i\right) \left[\Delta t \frac{\partial \mathrm{f}}{\partial \mathrm{t}} + \Delta w \frac{\partial \mathrm{f}}{\partial \mathrm{w}} \right] \right\} \tag{5.2-7}$$

which takes the form

$$a_0\left(i\right) \Delta w\left(i\right) = a_1\left(i\right) + \theta_2\left(i\right) \left\{ a_2\left(i\right) + \Delta t \left[\Delta p\left(ji\right) - \Delta p\left(jo\right) \right] + a_3\left(i\right) \Delta w\left(i\right) \right\} \tag{5.2-8}$$

or

$$\Delta w\left(i\right) = \frac{a_1\left(i\right) + \theta_2\left(i\right) \left\{ a_2\left(i\right) + \Delta t \left[\Delta p\left(ji\right) - \Delta p\left(jo\right) \right] \right\}}{a_0\left(i\right) - \theta_2\left(i\right) a_3\left(i\right)} \tag{5.2-9}$$

where ji and jo are the compressible volumes at the inlet and outlet of the liquid segment.

In general, the a's are sums of contributions from each element, k, in the segment. The terms are

$$a_0 = \sum_{\mathrm{k}} \Delta a_0\left(k\right) \tag{5.2-10}$$

$$\Delta a_0\left(k\right) = \frac{L_{\mathrm{k}}}{A_{\mathrm{k}}} \tag{5.2-11}$$

$$a_1 = \Delta t \left[p\left(ji, t\right) - p\left(jo, t\right) \right] - \Delta t \Delta p_{K_{in}} + \sum_{\mathrm{k}} \Delta a_1\left(k\right) \tag{5.2-12}$$

$$\Delta a_1\left(k\right) = \left[-\Delta p_{\mathrm{fr}}\left(k, t\right) - \Delta p_{\mathrm{gr}}\left(k, t\right) - \Delta p_{\mathrm{w2}}\left(k, t\right) - \Delta p_{\mathrm{v}}\left(k, t\right) + \Delta p_{\mathrm{p}}\left(k, t\right) \right] \Delta t \tag{5.2-13}$$

$$a_2 = \sum_{\mathrm{k}} \Delta a_2\left(k\right) \tag{5.2-14}$$

$$\Delta a_2\left(k\right) = \Delta t^2 \frac{\partial}{\partial \mathrm{t}} \left[\Delta p_{\mathrm{P}}\left(k\right) - \Delta p_{\mathrm{fr}}\left(k\right) - \Delta p_{\mathrm{w2}}\left(k\right) - \Delta p_{\mathrm{v}}\left(k\right) - \Delta p_{\mathrm{gr}}\left(k\right) \right] \tag{5.2-15}$$

$$a_3 = -\Delta t \frac{\partial \Delta p_{K_{in}}}{\partial w} + \sum_{\mathrm{k}} \Delta a_3\left(k\right) \tag{5.2-16}$$

and

$$\begin{aligned} \frac{\partial \Delta p_{K_{in}}}{\partial w} &= 2 \frac{\Delta p_{K_{in}}}{w} + \frac{\Delta p_{K_{in}}}{K_{in}(Re_{in})} \frac{\partial Re_{in}}{\partial \mathrm{w}} \frac{\partial K_{in}}{\partial Re_{in}} \\ \Delta a_3\left(k\right) &= \Delta t \frac{\partial}{\partial \mathrm{w}} \left[\Delta p_{\mathrm{p}}\left(k\right) - \Delta p_{\mathrm{fr}}\left(k\right) - \Delta p_{\mathrm{w2}}\left(k\right) - \Delta p_{\mathrm{v}}\left(k\right) - \Delta p_{\mathrm{gr}}\left(k\right) \right] \end{aligned} \tag{5.2-17}$$

In the above equations, it should be recognized that Δ is used in three different ways. First Δt is the time step, $\Delta w\left(i\right)$ is the change in the mass flow rate in the liquid segment i during the time step, and $\Delta p\left(ij\right)$ and $\Delta p\left(jo\right)$ are the changes in pressures in the compressible volumes at the inlet and outlet ends of liquid segment i during the time step. Second, the Δp's in Eq. (5.2-2) represent pressure differences, increases or decreases, along liquid segment i. And third, as seen in Eq. (5.2-10), Eq. (5.2-12), Eq. (5.2-14), and Eq. (5.2-16), the Δa's are incremental contributions form each of the elements to the a's for the whole segment.

The denominator in Eq. (5.2-9) should never be zero because a_3 is negative or zero, since friction increases with increased flow and pump head decreases with increased flow.

5.2.1.2 Average Flow Rate

The second main equation used in the hydraulics calculation is an expression for the average mass flow rate in a liquid segment during a time step. The average mass flow rate for segment i is taken as a simple average of the flow rate at the beginning of the time step and that at the end of the time step.

$$\overline{w}(i) = \frac{[w(i,t) + w(i,t+\Delta t)]}{2} \tag{5.2-18}$$

Linearization consists in expanding $w(i,t+\Delta t)$ to two terms and finite differencing the time derivative to give

$$\overline{w}(i) = \frac{2w(i,t) + \Delta w(i)}{2} \tag{5.2-19}$$

Eq. (5.2-19) relates the average mass flow rate in a segment during a time step to the change in the mass flow rate during that time step.

5.2.1.3 Compressible Volume Pressure Changes

The third main equation in the hydraulics calculations is an expression for the change in pressure in a compressible volume during a time step. The pressure in a compressible volume can be affected in several ways. Liquid can flow in or out through the segments connecting the compressible volumes. The entering liquid may be at a higher or lower temperature than that already there, and the liquid flowing out removes liquid at the compressible volume temperature. In addition to the changes related to the liquid flows, the compressible volumes can be heated or cooled externally, or liquid can be added or withdrawn by an external agent. The flow of cover gas into or out of the compressible volume will also affect the pressure. The cover gas flows are treated separately, as in Section 5.7.

The pressure in a compressible volume is assumed to vary linearly with changes in the mass or temperature of the liquid. Therefore, the change $\Delta p(j)$ in pressure in the compressible volume j during a time step is taken as a linear approximation in the average mass flow rates into and out from that compressible volume as:

$$\Delta p(j) = b_0(j) + b_1(j)\left[\sum \overline{w}_{\text{in}}(j) - \sum \overline{w}_{\text{out}}(j)\right] + b_2(j)\left[\sum \overline{w}_{\text{in}}(j)\,T_{\text{in}} - \sum \overline{w}_{\text{out}} T_{\text{out}}(j)\right] \tag{5.2-20}$$

Here $\sum \overline{w}_{\text{in}}(j)$ is the sum of the average mass flow rates into compressible volume j from all of the attached liquid segments flowing into it. It should be noticed that Eq. (5.2-19) is the expression for the average mass flow rate in a segment, and if that segment flows into compressible volume j, then it is included with all of the other segment contributions to compressible volume j. Similarly, $\sum \overline{w}_{\text{out}}(j)$ is the sum of the average mass flow rates out of the compressible volume j from all of the attached liquid segments flowing out. The last two sums in Eq. (5.2-20) are the same as the ones just described except that all of the mass flow rates are multiplied by the temperatures of the flows: the average mass flow rates flowing into the compressible volume are multiplied by the temperatures in the respective segments, whereas the average mass flow rates flowing out are each multiplied by the temperature in the compressible volume. The coefficients $b_1(j)$ and $b_2(j)$ include the time-step size and are computed for each type of compressible volume. The remaining term $b_0(j)$ also contains the time step size and can be used to account for the effects of heat transfer to the compressible volume liquid from the compressible volume wall or from other components.

5.2.2 Estimated Core Flow

The liquid segments representing the core channels are a special case. In principle, the coolant mass flow rates in the core channels could be calculated simultaneously along with the other segments in the primary loop. But after the onset of boiling in the core, this would unduly complicate the boiling model. Instead, an estimate of the core mass flow rates is made from information stored by the core channel coolant dynamics routines during the previous time step, and this estimated flow is included in the primary loop calculation. After the primary and intermediate loop hydraulics calculations have been done, the core channel coolant dynamics routines compute the actual channel flows for each channel independently, using the newly calculated inlet and outlet plenum pressures as boundary conditions. Then, the differences between the estimated core flow and the actual computed core flow for a time step is used to adjust the coolant masses in the inlet and outlet plenums before the start of the calculations for the next time step.

The mass flow rate for core channel ic at the end L, where $L = 1$ is the inlet and $L = 2$ is the outlet, is estimated by the momentum equation

$$\frac{dw_{\mathrm{c}}\left(L,ic\right)}{\mathrm{dt}} = C_0\left(L,ic\right) + C_1\left(L,ic\right)p\left(\mathrm{JIN}\right) + C_2\left(L,ic\right)p\left(\mathrm{JX}\right) + C_3\left(L,ic\right)dp_c\left(L,ic\right) \tag{5.2-21}$$

where $p\left(\mathrm{JIN}\right)$ is the pressure in the compressible volume representing the inlet plenum, $p\left(\mathrm{JX}\right)$ is the pressure in the compressible volume representing the outlet plenum, and $dp_c\left(L,ic\right)$ is the pressure drop from the core channel inlet to outlet. The rate of change of the mass flow rate is taken as proportional to the inlet and outlet pressures and the pressure drop across the core channel. The coefficients C_0, C_1, C_2, and C_3 are the information stored during the previous time step, and are described in Section 3.12 of Chapter 3. The channel mass flow rate $w_{\mathrm{c}}\left(L,ic\right)$ is evaluated at the beginning of the time step.

Eq. (5.2-21) can be written in finite difference form as

$$\begin{aligned}\frac{\Delta w_{\mathrm{c}}\left(L,ic\right)}{\Delta\mathrm{t}} = {} & C_0\left(L,ic\right) + C_1\left(L,ic\right)\left(\theta_{1c}p\left(\mathrm{JIN}\right) + \theta_{2c}\left(p\left(\mathrm{JIN}\right) + \Delta p\left(\mathrm{JIN}\right)\right)\right) \\ & + C_2\left(L,ic\right)\left(\theta_{1c}p\left(\mathrm{JX}\right) + \theta_{2c}\left(p\left(\mathrm{JX}\right) + \Delta p\left(\mathrm{JX}\right)\right)\right) \\ & + C_3\left(L,ic\right)\left(\theta_{1c}dp_c\left(L,ic\right) + \theta_{2c}\left(dp_c\left(L,ic\right) + \Delta dp_c\left(L,ic\right)\right)\right)\end{aligned} \tag{5.2-22}$$

By redefining $C_0\left(L,ic\right)$ and $C_3\left(L,ic\right)$,

$$C_0'\left(L,ic\right) = C_0\left(L,ic\right) + C_3 dp_c\left(L,ic\right) \tag{5.2-23}$$

and

$$C_3'\left(L,ic\right) = \frac{C_3\left(L,ic\right)}{2w_{\mathrm{c}}\left(L,ic\right)}\frac{\Delta dp_c\left(L,ic\right)}{\Delta w_{\mathrm{c}}\left(L,ic\right)} \tag{5.2-24}$$

Eq. (5.2-22) can be simplified to

$$\begin{aligned}\frac{\Delta w_{\mathrm{c}}\left(L,ic\right)}{\Delta\mathrm{t}} = {} & C_0'\left(L,ic\right) + C_1\left(L,ic\right)\left(p\left(\mathrm{JIN}\right) + \theta_{2c}\Delta p\left(\mathrm{JIN}\right)\right) \\ & + C_2\left(L,ic\right)\left(p\left(\mathrm{JX}\right) + \theta_{2c}\Delta p\left(\mathrm{JX}\right)\right) \\ & + 2\theta_{2c}C_3'\left(L,ic\right)w_{\mathrm{c}}\left(L,ic\right)\Delta w_{\mathrm{c}}\left(L,ic\right)\end{aligned} \tag{5.2-25}$$

or

$$\begin{aligned}\Delta w_{\mathrm{c}}\left(L,ic\right) = {} & \frac{\Delta t}{1 - 2\theta_{2\mathrm{c}}\left(L,ic\right)C_3'\left(L,ic\right)w_{\mathrm{c}}\left(L,ic\right)\Delta t}\left\{C_0'\left(L,ic\right)\right. \\ & + C_1\left(L,ic\right)\left[p\left(\mathrm{JIN},t\right) + \theta_{2\mathrm{c}}\left(L,ic\right)\Delta p\left(\mathrm{JIN}\right)\right] \\ & \left. + C_2\left(L,ic\right)\left[p\left(\mathrm{JX},t\right) + \theta_{2\mathrm{c}}\left(L,ic\right)\Delta p\left(\mathrm{JX}\right)\right]\right\}\end{aligned} \tag{5.2-26}$$

where θ_{2c} is the degree of implicitness for core channel ic. Eq. (5.2-26) plays the role for each core channel that Eq. (5.2-9) does for all the other liquid segments in the primary loop.

Before boiling begins in the core channels, the differences between the estimated and the actual core flows are very small, largely because Eq. (5.2-21) is equivalent to the equation used by the pre-boiling core channel coolant dynamics routines, except that the coefficients in Eq. (5.2-21) do not account for the effects of coolant temperature changes during the current time step. After boiling begins, rapid changes in vapor pressures cause rapid changes in the inlet plenum pressure, necessitating a decrease in the time step size.

5.2.3 Method of Solution

Eq. (5.2-9), Eq. (5.2-19), Eq. (5.2-20), and Eq. (5.2-26) constitute a set of simultaneous equations for changes in the mass flow rates $\Delta w\,(i)$ in the liquid segments and for changes in the pressures $\Delta p\,(j)$ in the compressible volumes during a time step. Eliminating the Δw's, these equations can be written as a single matrix equation for the Δp's:

$$\sum_{\mathrm{J}} c\,(I,J)\,\Delta p\,(J) = d\,(I) \tag{5.2-27}$$

The coefficients $c\,(I,J)$ and $d\,(I)$ are sums of contributions $\Delta c\,(I,J)$ and $\Delta d\,(I)$ from each segment. From segment i, in which the flow is from compressible volume I to compressible volume J, the contribution to $c\,(I,J)$ is

$$\Delta c\,(I,J) = \frac{-\theta_2\,(i)\,[b_1\,(J) + b_2\,(J)\,T_{\text{out}}\,(i)]\,\Delta t}{2\,[a_0\,(i) - a_3\,(i)\,\theta_2\,(i)]} \tag{5.2-28}$$

and the contribution to $c\,(J,I)$ is

$$\Delta c\,(J,I) = \frac{-\theta_2\,(i)\,[b_1\,(I) + b_2\,(I)\,T_{\text{in}}\,(i)]\,\Delta t}{2\,[a_0\,(i) - a_3\,(i)\,\theta_2\,(i)]} \tag{5.2-29}$$

The contribution to $d\,(J)$ is

$$\Delta d\,(J) = w\,(i,t) + \frac{a_1\,(i) + \theta_2\,(i)\,a_2\,(i)}{2\,[a_0\,(i) - \theta_2\,(i)\,a_3\,(i)]}\,[b_1\,(J) + b_2\,(J)\,T_{\text{out}}\,(i)] \tag{5.2-30}$$

and the contribution to $d\,(I)$ is

$$\Delta d\,(I) = -w\,(i,t) - \frac{a_1\,(i) + \theta_2\,(i)\,a_2\,(i)}{2\,[a_0\,(i) - \theta_2\,(i)\,a_3\,(i)]}\,[b_1\,(I) + b_2\,(I)\,T_{\text{in}}\,(i)] \tag{5.2-31}$$

The contributions to diagonal terms are

$$\Delta c\,(I,I) = \frac{\theta_2\,(i)\,[b_1\,(I) + b_2\,(I)\,T_{\text{in}}\,(i)]\,\Delta t}{2\,[a_0\,(i) - \theta_2\,(i)\,a_3\,(i)]} \tag{5.2-32}$$

and

$$\Delta c\,(J,J) = \frac{\theta_2\,(i)\,[b_1\,(J) + b_2\,(J)\,T_{\text{out}}\,(i)]\,\Delta t}{2\,[a_0\,(i) - \theta_2\,(i)\,a_3\,(i)]} \tag{5.2-33}$$

Also, $b_0\,(J)$ is added to $d\,(J)$ and 1.0 is added to the diagonal terms in c. In these equations $T_{\text{in}}\,(i)$ and $T_{\text{out}}\,(i)$ are the temperatures at the inlet and outlet of the liquid segment.

The contributions to the coefficients $c\,(I,J)$ and $d\,(I)$ from the segments representing the core channel flows are obtained from Eq. (5.2-26), which is re-written as

$$\Delta w_{\text{c}}\,(L,ic) = e_0\,(L,ic) + e_1\,(L,ic)\,\Delta p\,(\text{JIN}) + e_2\,(L,ic)\,\Delta p\,(\text{JX}) \tag{5.2-34}$$

where

$$e_0(L,ic) = \frac{\Delta t}{d_n}\left[C_0(L,ic) + C_1(L,ic)\,p(\text{JIN},t) + C_2(L,ic)\,p(\text{JX},t) + C_3(L,ic)\,w_c(L,ic)\,|w_c(L,ic)|\right] \tag{5.2-35}$$

$$e_1(L,ic) = \frac{\theta_{2c}(L,ic)\,\Delta t C_1(L,ic)}{d_n} \tag{5.2-36}$$

$$e_2(L,ic) = \frac{\theta_{2c}(L,ic)\,\Delta t C_2(L,ic)}{d_n} \tag{5.2-37}$$

and

$$d_n = 1 - 2\theta_{2c}(L,ic)\,C_3(L,ic)\,|w_c(L,ic)|\,\Delta t \tag{5.2-38}$$

The contribution to $d(\text{JIN})$ is then

$$\Delta d(\text{JIN}) = -\left[b_1(\text{JIN}) + b_2(\text{JIN})\,T_{\text{in}}(ic)\right]\left[w_c(1,ic) + \frac{e_0(1,ic)}{2}\right] \tag{5.2-39}$$

also,

$$\Delta d(\text{JX}) = \left[b_1(\text{JX}) + b_2(\text{JX})\,T_{\text{out}}(ic)\right]\left[w_c(2,ic) + \frac{e_0(2,ic)}{2}\right] \tag{5.2-40}$$

$$\Delta c(\text{JIN},\text{JIN}) = \left[b_1(\text{JIN}) + b_2(\text{JIN})\,T_{\text{in}}(ic)\right]\frac{e_1(1,ic)}{2} \tag{5.2-41}$$

$$\Delta c(\text{JX},\text{JX}) = -\left[b_1(\text{JIN}) + b_2(\text{JX})\,T_{\text{out}}(ic)\right]\frac{e_2(2,ic)}{2} \tag{5.2-42}$$

$$\Delta c(\text{JIN},\text{JX}) = \left[b_1(\text{JIN}) + b_2(\text{JIN})\,T_{\text{in}}(ic)\right]\frac{e_2(1,ic)}{2} \tag{5.2-43}$$

and

$$\Delta c(\text{JX},\text{JIN}) = \left[b_1(\text{JX}) + b_2(\text{JX})\,T_{\text{out}}(ic)\right]\frac{e_1(2,ic)}{2} \tag{5.2-44}$$

The contributions to the coefficients $c(I,J)$ and $d(J)$ from all the liquid segments in a loop have been made at this point, and Eq. (5.2-27) is solved by Gaussian elimination to yield the pressure changes in all the compressible volumes in the loop during the time step. With the pressure changes now known, Eq. (5.2-9) is solved for the mass flow-rate changes in the liquid segments in the loop, and Eq. (5.2-26) is solved for the estimated channel flow changes during the time step. The pressure changes and the mass flow rate changes are then added to the values of the pressures and flow rates at the beginning of the time step to obtain the respective values at the end of the time step.

The above procedure is carried out separately for the primary loops and for the intermediate loops. The core channel flow segments are included only in the primary loops.

5.2.4 Degree of Implicitness

As mentioned in Section 5.2.1, the parameters θ_1 and θ_2 determine the degree of implicitness of the calculation. For small time steps, a semi-implicit treatment with $\theta_1 = \theta_2 = .5$ is most accurate. For large time steps, a fully implicit calculation with $\theta_1 = 0$ and $\theta_2 = 1$ is more accurate and numerically more stable. As discussed in Section 3.19.1, the degree of implicitness is computed separately for each liquid segment, i, as

$$\theta_2(j) = \frac{a + b\gamma(i) + \gamma(i)^2}{2a + c\gamma(i) + \gamma(i)^2} \tag{5.2-45}$$

where

$$\gamma(i) = -\frac{a_3(i)}{a_0(i)} \tag{5.2-46}$$

$a = 6.12992$

$b = 2.66054$

and

$c = 3.56284$

Then

$$\theta_1(i) = 1 - \theta_2(i) \tag{5.2-47}$$

Note that $a_3(i)$ is always negative and is proportional to Δt, the time-step size. Eq. (5.2-45) and Eq. (5.2-46) give the results that θ_2 approaches 0.5 for small time steps, θ_2 approaches 1.0 for large time steps, and θ_2 makes a smooth transition between 0.5 and 1.0 for intermediate-sized time steps.

Liquid segments attached to an almost incompressible liquid junction are a special case. Because such compressible volumes are much smaller than other compressible volumes, their time constants for changes of all kinds are short; and because of it, these liquid segments are treated with a fully implicit ($\theta_2 = 1.0$) flow calculation regardless of time-step size.

The degree of implicitness used in the calculated estimated channel flow for channel ic is

$$\theta_{2c}(L, ic) = \frac{a + b\gamma_c(L, ic) + \gamma_c(L, ic)^2}{2a + c\gamma_c(L, ic) + \gamma_c(L, ic)^2} \tag{5.2-48}$$

where γ_c is given by

$$\gamma_c = \frac{2\Delta t}{w_c(L, ic)\, C_3(L, ic)} \tag{5.2-49}$$

5.3 Specific Components: Contributions to Flow and Pressure Equations

In this section, we shall consider the contributions to the mass flow-rate and pressure equations from the various components in the primary and intermediate loops. Each compressible volume contributes a b_0, a b_1, and a b_2 in Eq. (5.2-20), and each element in each segment connecting two compressible volumes contributes a Δa_0, a Δa_1, a Δa_2, and a Δa_3 in Eq. (5.2-11), Eq. (5.2-13), Eq. (5.2-15), and Eq. (5.2-17). We shall consider compressible volumes without cover gas, compressible volumes with cover gas, pipes, intermediate heat exchangers, and pumps.

5.3.1 Compressible Volumes Without Cover Gas

A compressible volume with no cover gas is treated as a compressible liquid in an expandable container. The volume V is assumed to vary linearly with pressure p and temperature T:

$$V = V_r\left[1 + \alpha_p(p - p_r) + \alpha_T(T - T_r)\right] \tag{5.3-1}$$

where V_r is the volume at a reference pressure p_r and reference temperature T_r. Also the coolant density ρ is assumed to vary linearly with p and T:

$$\rho = \rho_\mathrm{r}\left[1 + \beta_\mathrm{p}\left(p - p_\mathrm{r}\right) + \beta_\mathrm{T}\left(T - T_\mathrm{r}\right)\right] \tag{5.3-2}$$

where

α_p = the volume pressure expansion coefficient, $\frac{1}{V}\frac{\partial \mathrm{V}}{\partial \mathrm{p}}$

α_T = the volume thermal expansion coefficient, $\frac{1}{V}\frac{\partial \mathrm{V}}{\partial \mathrm{T}}$

β_p = the sodium compressibility, $\frac{1}{\rho}\frac{\partial \rho}{\partial \mathrm{p}}$

β_T = the sodium thermal expansion coefficient, $\frac{1}{\rho}\frac{\partial \rho}{\partial \mathrm{T}}$

The mass of the liquid in the compressible volume is

$$m = \rho V \tag{5.3-3}$$

Using Eq. (5.3-1) and Eq. (5.3-2) in Eq. (5.3-3) and dropping second-order terms gives

$$m = m_\mathrm{r}\left[1 + \left(\alpha_\mathrm{p} + \beta_\mathrm{p}\right)\left(p - p_\mathrm{r}\right) + \left(\alpha_\mathrm{T} + \beta_\mathrm{T}\right)\left(T - T_\mathrm{r}\right)\right] \tag{5.3-4}$$

which can be rewritten as

$$\delta p = \frac{\frac{\delta m}{m_\mathrm{r}} - \left(\alpha_\mathrm{T} + \beta_\mathrm{T}\right)\delta T}{\alpha_\mathrm{p} + \beta_\mathrm{p}} \tag{5.3-5}$$

where

$\delta m = m - m_\mathrm{r}$

$\delta p = p - p_\mathrm{r}$

$\delta T = T - T_\mathrm{r}$

Eq. (5.3-5) is a general relationship for the pressure change in a compressible volume with no cover gas as a result of mass and temperature changes.

To obtain expressions for $b_0\left(j\right)$, $b_1\left(j\right)$, and $b_2\left(j\right)$, which characterize this compressible volume j, we apply conservation of mass and conservation of energy principles to the volume with flow in and flow out during a time step. Conservation of mass gives

$$\Delta m = \Delta t\left[\sum \overline{w}_\mathrm{in} - \sum \overline{w}_\mathrm{out}\right] \tag{5.3-6}$$

where Δm is the change in the liquid mass in the compressible volume during the time step Δt, $\sum \overline{w}_\mathrm{in}$ is the sum of the average mass flow rates into the compressible volume during Δt, and $\sum \overline{w}_\mathrm{out}$ is the sum of the average mass flow rates out during Δt.

Conservation of energy gives

$$\left(m_3 + \Delta m\right)\left(T_3 - \Delta T\right) = m_3 T_3 + \Delta t\left[\sum \overline{w}_\mathrm{in} T_\mathrm{in} - \sum \overline{w}_\mathrm{out} T_\mathrm{out}\right] + \frac{Q}{c_\mathrm{l}}\Delta t \tag{5.3-7}$$

where m_3 and T_3 are the mass and temperature of the liquid in the compressible volume at the beginning of the time step, ΔT is the change in temperature of the liquid in the compressible volume during the time step, $\sum \overline{w}_\mathrm{in} T_\mathrm{in}$ in the sum of the average mass flow rates into the volume multiplied by the incoming temperature, $\sum \overline{w}_\mathrm{out} T_\mathrm{out}$ is the sum of the average mass flow rates out of the volume multiplied by the outgoing temperature, Q is the heat flow rate from the compressible volume walls and from other components in contact with the

compressible volume liquid, and c_l is the heat capacity of the liquid in the compressible volume. Eq. (5.3-7) expresses the fact that the energy in the liquid in the compressible volume at the end of the time step is the sum of the energy present at the beginning of the time step, the excess of the energy flowing in over that flowing out during the time step, and the energy contributed to the liquid from the walls of the compressible volume during the time step.

Solving Eq. (5.3-7) for the change in the liquid temperature during the time step, gives

$$\Delta T = \frac{-T_3 \Delta m + \Delta t \left[\sum \overline{w}_{in} T_{in} - \sum \overline{w}_{out} T_{out}\right] + \frac{Q}{c_l} \Delta t}{m_3 + \Delta m} \tag{5.3-8}$$

To first order, the $m_3 + \Delta m$ in the denominator is approximated as m_3. Inserting Eq. (5.3-6) and Eq. (5.3-8) into Eq. (5.3-5) gives

$$\begin{aligned} \Delta p = & \left[\sum \overline{w}_{in} - \sum \overline{w}_{out}\right] \frac{\Delta t}{\alpha_p + \beta_p} \left[\frac{1}{m_r} + \frac{(\alpha_T + \beta_T) T_3}{m_3}\right] \\ & - \frac{(\alpha_T + \beta_T) \Delta t}{(\alpha_p + \beta_p) m_3} \left[\sum \overline{w}_m T_{in} - \sum \overline{w}_{out} T_{out} + \frac{Q}{c_t}\right] \end{aligned} \tag{5.3-9}$$

Comparison of this equation with Eq. (5.2-20) shows that for the compressible volume with no cover gas

$$b_0 = -\frac{(\alpha_T + \beta_T) Q \Delta t}{(\alpha_p + \beta_p) m_3 c_l} \tag{5.3-10}$$

$$b_1 = \frac{\Delta t}{\alpha_p + \beta_p} \left[\frac{1}{m_r} + \frac{(\alpha_T + \beta_T) T_3}{m_3}\right] \tag{5.3-11}$$

and

$$b_2 = -\frac{(\alpha_T + \beta_T) \Delta t}{(\alpha_p + \beta_p) m_3} \tag{5.3-12}$$

The $Q\Delta t$ term in Eq. (5.3-10) is calculated in a manner similar to that described in Section 5.4.4, except that the value for b_0 is calculated before the temperatures at the end of the step are calculated, so q is calculated on the basis of temperatures at the beginning of the time step. In the present version of the code, the reference mass m_r and the reference temperature T_r are taken as the mass and temperature at the beginning of the time step.

5.3.2 Compressible Volumes With Cover Gas

A compressible volume with cover gas is treated in a fashion similar to that without cover gas except that the compression of the liquid is neglected compared with the compression of the gas, so that all of the expansion or compression is attributed entirely to the gas. The cover gas is assumed to expand or compress due to changes in volume, temperature, and mass. These changes are assumed to occur independently of each other. An increase in the gas volume is equal to the decrease in the liquid volume. The accompanying rise or fall in the level of the gas-liquid interface is taken as the volume change divided by the area of the compressible volume. In addition, the conservation of mass and the conservation of energy principles are observed. The conservation of mass is applied by taking the increase in the mass of the liquid in a compressible volume as the difference between the liquid flowing in and that flowing out during a time step, and the conservation of energy is taken as the increase in the mass of the liquid in the compressible volume times its temperature as the difference between the mass times temperature flowing in and that flowing out during a time step.

The liquid pressure at an elevation, z_r, in the compressible volume is given by

$$p_l = p_g + \rho_l g \left(z_i - z_r\right) \tag{5.3-13}$$

where

p_{l} = the pressure in the liquid

p_{g} = the pressure of the cover gas above the liquid

ρ_{l} = the liquid density

g = the acceleration of gravity

z_{i} = the height of the liquid gas interface

z_{r} = the reference height for the compressible volume, **ZCVL**

and the change in the liquid pressure in the compressible volume is obtained by taking differentials of Eq. (5.3-13):

$$\Delta p_{\mathrm{l}} = \Delta p_{\mathrm{g}} + g\left(z_{\mathrm{i}} - z_{\mathrm{r}}\right)\Delta\rho_{\mathrm{l}} + \rho_{\mathrm{l}} g \Delta z_{\mathrm{i}} \tag{5.3-14}$$

5.3.2.1 Cover Gas Contribution

Adiabatic compression of the cover gas is taken as

$$p_{\mathrm{g}} V_{\mathrm{g}}^{\gamma} = \text{const} \tag{5.3-15}$$

where

V_{g} = the volume of the cover gas

γ = the ratio of the specific heat at constant pressure to that at constant volume for the cover gas

and in differential form becomes:

$$\frac{\Delta p_{\mathrm{g,v}}}{p_{\mathrm{g}}} + \gamma \frac{\Delta V_{\mathrm{g}}}{V_{\mathrm{g}}} = 0 \tag{5.3-16}$$

where $\Delta p_{\mathrm{g,v}}$ is the change in gas pressure due to the change in gas volume. To account for the change in gas pressure due to the change in gas temperature, Eq. (5.7-3) can be used to determine the change in gas temperature:

$$\Delta T_{\mathrm{g}} = \Delta t \frac{T_{\mathrm{l}} - T_{\mathrm{g}}}{\Delta t + \tau} \tag{5.3-17}$$

where

T_{g} = cover gas temperature,

τ = cover gas temperature time constant, **TAUGAS**.

Using the ideal gas law:

$$\frac{p_{\mathrm{g}}}{T_{\mathrm{g}}} = \text{const} \tag{5.3-18}$$

$$\Delta p_{\mathrm{g,t}} = \frac{p_{\mathrm{g}} \Delta T_{\mathrm{g}}}{T_{\mathrm{g}}} = \frac{p_{\mathrm{g}} \Delta t \left(T_{\mathrm{l}} - T_{\mathrm{g}}\right)}{T_{\mathrm{g}}\left(\Delta t + \tau\right)} \tag{5.3-19}$$

where $\Delta p_{\mathrm{g,t}}$ is the change in gas pressure due to the change in gas temperature. To account for the change in gas pressure due to the change in gas mass, Eq. (5.7-14) can be used:

$$\Delta p_{\mathrm{g,m}} = p_{\mathrm{g}} \varepsilon \tag{5.3-20}$$

$$\varepsilon = \frac{\gamma \Delta t \frac{\mathrm{d}m_{\mathrm{g3}}}{\mathrm{dt}}}{m_{\mathrm{g}}} \tag{5.3-21}$$

where $\frac{\mathrm{d}m_{\mathrm{g3}}}{\mathrm{dt}}$ is from the previous time step. At each time step, the new value is calculated as

$$\frac{\mathrm{d}m_{\mathrm{g4}}}{\mathrm{dt}} = \frac{\Delta t_{\max}}{\Delta t_{\max} + \Delta t} \frac{\mathrm{d}m_{\mathrm{g3}}}{\mathrm{dt}} + \frac{\Delta t}{\Delta t_{\max} + \Delta t} \frac{\Delta m_{\mathrm{g}}}{\Delta t} \tag{5.3-22}$$

where $\Delta t_{\max}$ is **DTPMAX**. Finally, let

$$\Delta p_{\mathrm{g,m}} = \frac{p_{\mathrm{g}} \varepsilon}{1 + 10\varepsilon^2} \tag{5.3-23}$$

Eq. (5.3-22) and Eq. (5.3-23) are used to dampen oscillations caused by the first order approximations. Combining the previous equations:

$$\Delta p_{\mathrm{g}} = \Delta p_{\mathrm{g,v}} + \Delta p_{\mathrm{g,t}} + \Delta p_{\mathrm{g,m}} \tag{5.3-24}$$

5.3.2.2 Liquid Volume Contribution

The conservation of liquid mass for a compressible volume gives

$$\Delta m_{\mathrm{l}} = \Delta t \left[\sum \overline{w}_{\mathrm{in}} - \sum \overline{w}_{\mathrm{out}} \right] \tag{5.3-25}$$

where

Δm_{l} = the liquid mass increase in the compressible volume during the time step

Δt = the time-step size

$\sum \overline{w}_{\mathrm{in}}$ = the sum of the average liquid mass flow rates into the compressible volumes during the time step

$\sum \overline{w}_{\mathrm{out}}$ = the sum of the average liquid mass flow rates out from the compressible volume during the time step.

The conservation of energy for a compressible volume yields

$$(m_{\mathrm{l}} + \Delta m_{\mathrm{l}})(T_{\mathrm{l}} + \Delta T_{\mathrm{l}}) = m_{\mathrm{l}} T_{\mathrm{l}} + \Delta t \left[\sum \overline{w}_{\mathrm{in}} T_{\mathrm{in}} - \sum \overline{w}_{\mathrm{out}} T_{\mathrm{out}} \right] + \frac{(T_{\mathrm{w}} - T_{\mathrm{l}}) H_{\mathrm{w}} A_{\mathrm{w}} \Delta t}{C_{\mathrm{p}}} \tag{5.3-26}$$

where

m_{l} = the liquid mass at the beginning of the time step

ΔT_{l} = the increase in liquid temperature in the compressible volume during the time step

$\sum \overline{w}_{\mathrm{in}} T_{\mathrm{in}}$ = the sum of the average liquid mass flow rates times temperatures entering the compressible volume during the time step

$\sum \overline{w}_{\mathrm{out}} T_{\mathrm{out}}$ = sum of the average liquid mass flow rates times temperatures leaving the compressible volume during the time step

T_{w} = the compressible volume wall temperature at the beginning of the time step

H_{w} = the compressible volume wall-coolant heat-transfer coefficient, **HWALL**

A_{w} = the compressible volume wall surface area, **AWALL**

C_{p} = the liquid specific heat.

In the present version of the code, Δm_{l} is neglected in comparison with m, giving

$$\Delta T_{\mathrm{l}} = \frac{\Delta t}{m_{\mathrm{l}}} \left[\sum \overline{w}_{\mathrm{in}} T_{\mathrm{in}} - \sum \overline{w}_{\mathrm{out}} T_{\mathrm{out}} \right] + \Delta T_{\mathrm{wc}} \tag{5.3-27}$$

where

ΔT_{wc} = liquid temperature change due to heat transfer from wall to coolant, $\frac{(T_{\mathrm{w}} - T_{\mathrm{l}}) H_{\mathrm{w}} A_{\mathrm{w}} \Delta t}{m_{\mathrm{l}} C_{\mathrm{p}}}$

5.3.2.3 Numerical Implementation

In addition to the above equations, we take

$$\Delta V_{\mathrm{g}} = -\Delta V_{\mathrm{l}} \tag{5.3-28}$$

$$V_{\mathrm{l}} = \frac{m_{\mathrm{l}}}{\rho_{\mathrm{l}}} \tag{5.3-29}$$

$$\Delta z_{\mathrm{i}} = \frac{\Delta V_{\mathrm{l}}}{A} \tag{5.3-30}$$

where

V_{l} = the volume of the liquid in the compressible volume at the beginning of a time step

m_{l} = the mass of the liquid in the compressible volume at the beginning of a time step

ρ_{l} = the density of the liquid

A = the area of the liquid-gas interface in the compressible volume, **AREAIN**.

Differencing Eq. (5.3-29), we have

$$\Delta V_{\mathrm{l}} = \frac{\Delta m_{\mathrm{l}}}{\rho_{\mathrm{l}}} - \frac{m_{\mathrm{l}}}{\rho_{\mathrm{l}}^2}\Delta\rho_{\mathrm{l}} \tag{5.3-31}$$

and taking

$$\Delta\rho_{\mathrm{l}} = \frac{\partial\rho}{\partial \mathrm{T}}\Delta T_{\mathrm{l}} \tag{5.3-32}$$

we can write the following expression for the change in the liquid pressure during a time step as

$$\Delta p_{\mathrm{l}} = \left(\gamma\frac{p_{\mathrm{g}}}{V_{\mathrm{g}}} + \frac{\rho_{\mathrm{l}} g}{A}\right)\left(\frac{\Delta m_{\mathrm{l}}}{\rho_{\mathrm{l}}} - \frac{m_{\mathrm{l}}}{\rho_{\mathrm{l}}^2}\frac{\partial\rho}{\partial \mathrm{T}}\Delta T_{\mathrm{l}}\right) + \Delta p_{\mathrm{g,t}} + \Delta p_{\mathrm{g,m}} + g\left(z_{\mathrm{i}} - z_{\mathrm{r}}\right)\frac{\partial\rho}{\partial \mathrm{T}}\Delta T_{\mathrm{l}} \tag{5.3-33}$$

Inserting Eq. (5.3-25) for $\Delta m_{\mathrm{l}}$and Eq. (5.3-27) for ΔT_{l} and then comparing with Eq. (5.2-20), we see that the values of the b's for a compressible volume with a cover gas are

$$b_0 = \begin{cases} 0 & \text{if pipe rupture source} \\ \Delta p_{\mathrm{g,t}} & \text{if pipe rupture sink and } z_{\mathrm{i}} \le z_{\mathrm{r}} \\ \Delta p_{\mathrm{g,t}} + \Delta p_{\mathrm{g,m}} - g\left(z_{\mathrm{r}} - z_{\mathrm{i}}\right)\frac{\partial\rho}{\partial \mathrm{T}}\Delta T_{\mathrm{wc}} - \Delta z_{\mathrm{i}}\left(\rho g + \frac{A\gamma p_{\mathrm{g}}}{V_{\mathrm{g}}}\right) & \text{if pressurizer or pump bowl} \\ \Delta p_{\mathrm{g,t}} + \Delta p_{\mathrm{g,m}} & \text{otherwise} \end{cases} \tag{5.3-34}$$

$$b_1 = \begin{cases} 0 & \text{if pipe rupture source} \\ \Delta t\left[\frac{\gamma p_{\mathrm{g}}}{V_{\mathrm{g}}\rho}\right] & \text{if pipe rupture sink and } z_{\mathrm{i}} \le z_{\mathrm{r}} \\ \Delta t\left[\frac{\gamma p_{\mathrm{g}}}{V_{\mathrm{g}}\rho} + \frac{g}{A}\right] & \text{otherwise} \end{cases} \tag{5.3-35}$$

$$b_2 = \begin{cases} 0 & \text{if pipe rupture source or sink with } z_{\mathrm{i}} \le z_{\mathrm{r}} \\ \Delta t\frac{\partial\rho}{\partial \mathrm{T}}\left[\frac{g(z_{\mathrm{i}} - z_{\mathrm{r}})}{m_{\mathrm{l}}} + \frac{1}{\rho}\left(\frac{\gamma p_{\mathrm{g}}}{V_{\mathrm{g}}\rho} + \frac{g}{A}\right)\right] & \text{otherwise} \end{cases} \tag{5.3-36}$$

b_0 accounts for the change in pressure due to changes in gas mass and temperature. Different CV types have different contributions to b_0. The pressurizer (**ITYPCV** = 9) and pump bowl (**ITYPCV** = 10) CVs consider heat transfer from the wall to the coolant which impacts the coolant level.

The pipe rupture sink CV (**ITYPCV** = 6) assumes that the gas mass does not change (i.e. the volume is a closed container) and that the liquid level and temperature do not change when the liquid gas interface height is below the reference height (i.e. the break location). These assumptions simplify the b_0 and b_1 coefficients, and result in a b_2 coefficient of zero. If the liquid gas interface height is higher than the reference height, the pipe rupture sink is treated as a pool.

Lastly, the pipe rupture source CV (**ITYPCV** = 5) is assumed to maintain a constant pressure and temperature, so all of the b coefficients are zero.

5.3.2.4 Common Cover Gas

A CV with cover gas can share a common cover gas with another CV (see **ICCVFS** and **NCCV**). For a pool or outlet plenum with common cover gas, it is assumed that the total mass and volume available to the cover gas does not change. For a pressurizer or pump bowl with common cover gas, it is assumed that the total mass available to the cover gas does not change. The values of the b's for a compressible volume with common cover gas are

$$b_0^* = \begin{cases} 0 & \text{if pipe rupture source} \\ \Delta p_{\mathrm{g,t}} - g\left(z_{\mathrm{r}} - z_{\mathrm{i}}\right)\frac{\partial \rho}{\partial \mathrm{T}}\Delta T_{\mathrm{wc}} - \Delta z_{\mathrm{i}}\left(\rho g + \frac{A\gamma p_{\mathrm{g}}}{V_{\mathrm{g}}}\right) & \text{if pressurizer or pump bowl} \\ \Delta p_{\mathrm{g,t}} & \text{otherwise} \end{cases} \tag{5.3-37}$$

$$b_0 = \frac{\sum_{\mathrm{i=CCG1}}^{\mathrm{CCGN}} m_{\mathrm{g},i} b_{0,i}^*}{\sum_{\mathrm{i=CCG1}}^{\mathrm{CCGN}} m_{\mathrm{g},i}} + \gamma \Delta t p_{\mathrm{g}} \frac{\sum_{\mathrm{i=CCG1}}^{\mathrm{CCGN}} \frac{\mathrm{d}m_{\mathrm{g},i}}{\mathrm{dt}}}{\sum_{\mathrm{i=CCG1}}^{\mathrm{CCGN}} m_{\mathrm{g},i}} \tag{5.3-38}$$

where CCG1 is the first CV with common cover gas (**ICCVFS**) and CCGN is the last (**ICCVFS** + **NCCV**). b_0^* is calculated for each CV that shares a common cover gas. CVs with common cover gas have the same b_0 which is calculated using Eq. (5.3-38).

$$b_1 = \begin{cases} 0 & \text{if pipe rupture source} \\ \Delta t\left[\frac{\gamma p_{\mathrm{g}}}{V_{\mathrm{g}}\rho}\right] & \text{if pipe rupture sink and } z_{\mathrm{i}} \le z_{\mathrm{r}} \\ \Delta t\left[\frac{\gamma p_{\mathrm{g}}}{V_{\mathrm{g}}\rho} + \frac{g}{A}\right] & \text{if pressurizer or pump bowl} \\ \Delta t\left[\frac{g}{A}\right] & \text{otherwise} \end{cases} \tag{5.3-39}$$

$$b_2 = \begin{cases} 0 & \text{if pipe rupture source or sink with } z_{\mathrm{i}} \le z_{\mathrm{r}} \\ \Delta t\frac{\partial \rho}{\partial \mathrm{T}}\left[\frac{g(z_{\mathrm{i}} - z_{\mathrm{r}})}{m_{\mathrm{l}}} + \frac{1}{\rho}\left(\frac{\gamma p_{\mathrm{g}}}{V_{\mathrm{g}}\rho} + \frac{g}{A}\right)\right] & \text{if pressurizer or pump bowl} \\ \Delta t\frac{\partial \rho}{\partial \mathrm{T}}\left[\frac{g(z_{\mathrm{i}} - z_{\mathrm{r}})}{m_{\mathrm{l}}} + \frac{1}{\rho}\left(\frac{g}{A}\right)\right] & \text{otherwise} \end{cases} \tag{5.3-40}$$

5.3.3 Pipes and Intermediate Heat Exchangers

5.3.3.1 Contributions to Flow and Pressure

In considering the flow through a pipe or through an intermediate heat exchanger, several factors are taken into consideration. The contribution of the element to a_{o} in Eq. (5.2-10) is taken as

$$\Delta a_{\mathrm{o}} = L/A \tag{5.3-41}$$

where

L = the length of the element

A = the flow area of the element

The pressure drop contribution of the element to a_1 in Eq. (5.2-12) is composed of a number of terms. One of these terms is the frictional pressure drop, which is written as

$$\Delta p_{\text{fr}} = f \frac{L}{D_{\text{h}}} \frac{\rho v \left| v \right|}{2} = f \frac{L}{D_{\text{h}}} \frac{w \left| w \right|}{2 \rho A^2} \tag{5.3-42}$$

where

Δp_{fr} = the frictional pressure drop

f = the Moody friction factor

L = the length of the element

D_{h} = the hydraulic diameter of the element

ρ = the density of the liquid

v = the liquid velocity

w = the liquid mass flow rate

A = the flow area of the element

The Moody friction factor f for turbulent flow in pipes [5-2] is taken as

$$f = C_1 \left[1 + \left(C_2 \frac{\varepsilon}{D_{\text{h}}} + \frac{C_3}{\text{Re}} \right)^{C_4} \right] \tag{5.3-43}$$

where

$C_1 = 0.0055$

$C_2 = 20,000$

$C_3 = 1.0 \times 10^6$

$C_4 = 1/3$

ε = the user-supplied roughness of the element

Re = the Reynolds number

For laminar flow, the friction factor f is taken as

$$f = 64/\text{Re} \tag{5.3-44}$$

The Reynolds number in either case is

$$Re = \frac{D_{\text{h}} \left| w \right|}{A \mu} \tag{5.3-45}$$

where μ is the viscosity of the fluid.

A second term is the pressure drop caused by bends in the flow path, and these are modeled as an additional frictional drop term, written as

$$\Delta p_b = f \frac{L_{\text{B}}}{D_{\text{B}}} N_{\text{B}} \frac{w \left| w \right|}{2 \rho A^2} \tag{5.3-46}$$

where

L_B/D_B = a user-supplied input number for an effective length-to-diameter ratio per bend

N_B =the number of bends in the element

A third term is the pressure drop to account for baffles or restrictive orifices within the element which cause greater pressure drops than would be accounted for by friction or bends above. This term is taken as

$$\Delta p_{G2} = G_2' \frac{w\,|w|}{2\rho A^2} \tag{5.3-47}$$

where G_2' is the minor loss coefficient, defined as

$$G_2' = \begin{cases} G_2 & \text{if } Re \geq 100 \\ G_2\left(\frac{100}{Re}\right) & \text{if } Re < 100 \end{cases} \tag{5.3-48}$$

based on the user-provided nominal loss coefficient, G_2 (**G2PRDR**).

A forth term is the minor pressure loss that occurs at the outlet of an element. This term is defined as

$$\Delta p_{K_{out}} = K_{out}\left(Re_{out}\right) \frac{w\,|w|}{2\rho A_{out}^2} \tag{5.3-49}$$

where $K_{out}\left(Re_{out}\right)$ is the anisotropic Re-dependent loss coefficient at the outlet of the element, Re_{out} is the Re number at the outlet of the elment and A_{out} is the area at the outlet of the element. Anisotropic Re-dependent loss coefficients are described in Section 5.3.3.2.

A fifth item is the second term in Eq. (5.2-1), which is proportional to the square of the mass flow rate, and is due to the difference in the fluid densities at the inlet and outlet ends of the element. It has the form

$$\Delta p_A = \frac{w^2}{A^2}\left(\frac{1}{\rho_O} - \frac{1}{\rho_I}\right) \tag{5.3-50}$$

where

ρ_O = the fluid density at the element outlet

ρ_I = the fluid density at the element inlet

A final term is the gravity-head term for the element, taken as

$$\Delta p_{gr} = g\left(z_{\text{out}} - z_{\text{in}}\right)\overline{\rho} \tag{5.3-51}$$

where

z_{out} = the height of the element outlet

z_{in} = the height of the element inlet

g = the acceleration of gravity

$\overline{\rho}$ = the average of the inlet and outlet fluid densities

Taking the above five terms together, the contribution from the element to a_1 is

$$\Delta a_1 = -\Delta t\left[\Delta p_{fr} + \Delta p_b + \Delta p_{G2} + \Delta p_{K_{out}} + \Delta p_A + \Delta p_{gr}\right] \tag{5.3-52}$$

The contribution of the element to a_2, which is the derivative of the pressure drops with respect to time, as shown in Eq. (5.2-15), is zero, since the friction factors and the geometry are assumed not to change independently with time. Hence

$$\Delta a_2 = 0 \tag{5.3-53}$$

The contribution of the element to a_3, however, which is the derivative of the pressure drops with respect to the mass flow rate, as shown in Eq. (5.2-17), is obtained by differentiation. For laminar flow, differentiating Eq. (5.3-44) gives

$$\frac{\partial \mathrm{f}}{\partial \mathrm{w}} = -\frac{f}{w} \tag{5.3-54}$$

For turbulent flow, differentiating Eq. (5.3-43) gives

$$\frac{\partial \mathrm{f}}{\partial \mathrm{w}} = -\frac{C_1 C_3 C_4}{\mathrm{Re}\,|w|}\left(C_2\frac{\varepsilon}{D_\mathrm{h}} + \frac{C_3}{\mathrm{Re}}\right)^{C_4-1} \tag{5.3-55}$$

Taking the derivative with respect to w of each term in Eq. (5.3-52), we have for the contribution to a_3

$$\begin{aligned}\Delta a_3 = -\Delta t \Bigg\{ & 2\frac{\Delta p_{fr} + \Delta p_b + \Delta p_{G2} + \Delta p_{K_{out}} + \Delta p_A}{w} \\ & + \frac{\Delta p_{fr} + \Delta p_b}{f}\frac{\partial \mathrm{f}}{\partial \mathrm{w}} + \frac{\Delta p_{K_{out}}}{K_{out}(Re_{out})}\frac{\partial Re_{out}}{\partial \mathrm{w}}\frac{\partial K_{out}}{\partial Re_{out}}\Bigg\}\end{aligned} \tag{5.3-56}$$

5.3.3.2 Anisotropic Re-dependent Loss Coefficients

Anisotropic Re-dependent loss coefficients can be defined at element interfaces using Interface Loss Coefficient Tables. The definition of the loss coefficient tables is shown in Table 5.3.1, and examples are shown in Listing 5.3.1 and Listing 5.3.2. Loss coefficient tables are assigned to the outlet of an element using **`EllLossCoefTableID`** and to the inlet of a segment using **`SegLossCoefTableID`**,

By default, loss coefficients at element and segment interfaces are zero. Interfaces assigned to an interface loss coefficient table will overwrite the default interface loss coefficient. The pressure drop at element interface i caused by the loss coefficient is defined as

$$P_{i+} - P_{i-} = K_i(Re_i)\frac{w|w|}{2\rho_i A_i^2} \tag{5.3-57}$$

where

$$Re_i = \frac{Dh_i|w|}{A_i\mu_i} \tag{5.3-58}$$

and P_{i+}/P_{i-} are the pressure downstream and upstream the interface, respectively. In PRIMAR4, the pressure drop at an interface is included in the pressure drop of the element that defines the interface. In the case of a segment inlet interface, the pressure drop is included in the pressure drop of the first element of the segment.

Two functional forms are currently available to capture the dependence of the loss coefficient on the Reynolds number:

$$K_i(Re_i) = \begin{cases} C_1 + C_2 Re_i^{C_3} & \text{Kind} = 1 \\ C_1 + C_2 exp\,(Re_i C_3) & \text{Kind} = 2 \end{cases} \tag{5.3-59}$$

$$\begin{aligned} C_1 &\geq 0 \\ C_2 &\geq 0 \qquad \text{Kind} = 1 \\ C_3 &\geq -2 \end{aligned} \tag{5.3-60}$$

$$\begin{aligned} C_1 &\geq 0 \\ C_1 + C_2 &\geq 0 \end{aligned} \qquad \text{Kind} = 2$$

Two sets of coefficients may be defined for a given interface: one set for forward flow and one set for reverse flow. Additionally, an interface area, A_i, and hydraulic diameter, Dh_i, may be defined for the evaluation of the interface pressure drop and interface Re. If an interface area and interface hydraulic diameter are not provided or are defined as zero, the area and hydraulic diameter of the element upstream or downstream the interface are used. If the upstream element has a smaller or equal area than the downstream element, the upstream element's area and hydraulic diameter are used for the interface. If the upstream element has a bigger area than the downstream element, the downstream element's area and hydraulic diameter are used for the interface.

At very low flow rates, the derivative of the loss coefficients may approach infinity. This introduces numerical difficulty when SAS attempts to reverse flow through an element. In order to improve the robustness of the semi-implicit solution scheme, an optional lower limit has been introduced for Re-dependent loss coefficients, ReL. When the Reynold number drops below ReL, Eq. (5.3-59) is evaluated at ReL. Additionally, the derivative of $K_i(Re_i)$ is assumed to be zero.

Table 5.3.1: Input in a Interface Loss Coefficient Table

Column Label	Description	Range
`iEll` (Use only with element outlet table)	Index of the element outlet where the coefficient will be defined.	$1 \leq$ `iEll` $\leq$ **NELEMT**
`iSGL` (Use only with segment inlet table)	Index of the segment inlet where the coefficient will be defined.	$1 \leq$ `iSGL` $\leq$ **NSEGLP** + **NSEGLS**
`Kind`	Kind of functional form for calculating the anisotropic loss coefficient. See Eq. (3.9-24)	1 or 2
`C1f`, `C2f`, `C3f`	Function coefficients for forward flow	See Eq. (3.9-25)
`C1b`, `C2b`, `C3b`	Function coefficients for backward flow	See Eq. (3.9-25)
`A` (optional)	Area used to evaluate the Reynolds number and minor pressure loss at the interface. Optional area and hydraulic diameter must be provided as a pair.	$A \geq 0$
`Dh` (optional)	Hydraulic Diameter used to evaluate the Reynolds number at the interface. Optional area and hydraulic diameter must be provided as a pair.	$Dh \geq 0$
`ReL` (optional)	Lower Reynolds number limit for evaluating the functional form. For Reynolds numbers below the lower limit, the functional form is evaluated at *ReL* and the derivative is assumed to be 0.0.	$ReL \geq 0$

Listing 5.3.1: Example of Element Outlet Anisotropic Loss Coefficient Input

```
TABLE <id> <name>
iEll  Kind  C1f    C2f        C3f    C1b     C2b      C3b
2     1     0.5 0.0001          1    1.5 100000       -1
3     2     1        10 -0.000001      2 0.0001 0.00001
4     1     1   100000         -1    2.2  20000       -1
END
```

Listing 5.3.2: Example of Segment Inlet Anisotropic Loss Coefficient Input

```
TABLE <id> <name>
iSGL  Kind  C1f    C2f       C3f   C1b    C2b     C3b   A             Dh  ReL
1     1     0.5 100000        -1   1.5      2       0  2.2           6.5  0.0
4     1     1   100000        -1   2.2  20000      -1 0.002 0.003183099  1.0
END
```

5.3.4 Pumps

Pump components are an extension of the pipe components, in terms of hydraulic contributions. Pump components add the impact of the pump head to the pipe hydraulic contributions, Δa_0^{pipe}, Δa_1^{pipe}, Δa_2^{pipe}, and Δa_3^{pipe}, defined in Eq. (5.3-41), Eq. (5.3-52), Eq. (5.3-53) and Eq. (5.3-56), respectively. There are three pump choices available in the PRIMAR-4 module: a *user-supplied table look-up*, a *centrifugal pump*, and an *electromagnetic pump*. There are a number of options for the centrifugal and electromagnetic pumps.

5.3.4.1 Table Look-Up

The user supplies a table of relative pump head vs. time, and the code interpolates linearly between points in the table. The pump head as a function of time is given by

$$H(t) = H_{\text{r}} f(t) \tag{5.3-61}$$

where $f(t)$ is the user-supplied table, with $f(t=0) = 1.0$, and H_{r} is the reference pump head for the particular pump. The contribution to the coefficients in Eq. (5.2-11), Eq. (5.2-13), Eq. (5.2-15), and Eq. (5.2-17) are

$$\Delta a_0 = \Delta a_0^{\text{pipe}} \tag{5.3-62}$$

$$\Delta a_1 = \Delta t H(t_3) + \Delta a_1^{\text{pipe}} \tag{5.3-63}$$

$$\Delta a_2 = \Delta t \left[H(t_4) - H(t_3)\right] + \Delta a_2^{\text{pipe}} \tag{5.3-64}$$

$$\Delta a_3 = \Delta a_3^{\text{pipe}} \tag{5.3-65}$$

where

Δt = the time interval

t_3, t_4 = the times at the beginning and end of the time interval

5.3.4.2 Centrifugal Pumps

5.3.4.2.1 Option 1

In this option, the pump head and the pump torque are represented either by constant values or by polynomial fits, depending upon the pump flow rates or upon the pump speeds. The flow rates and speeds, and also the pump head and torque, are expressed as multiples of rated quantities, which are user-supplied. The constant values and the coefficients in the polynomial fits are also user-supplied.

The pump head and torque are expressed as functions of the parameter χ, which is

$$\chi = \frac{\overline{w}}{\overline{s}} \tag{5.3-66}$$

with

$$\overline{w} = \frac{w}{w_{\mathrm{R}}} \tag{5.3-67}$$

$$\overline{s} = \frac{s}{s_{\mathrm{R}}} \tag{5.3-68}$$

where

w = the pump flow rate

w_{R} = the rated pump flow rate

s = the pump speed

s_{R} = the rated pump speed

Pump Head

For positive pump speeds the pump head is taken as

$$\Delta p_{\mathrm{p}} = H_{\mathrm{R}} s^2 \left(A_1 + A_2\chi + A_3\chi^2 + A_4\chi^3 + A_5\chi^4\right), \quad \text{if } \chi < A_6 \tag{5.3-69}$$

$$\Delta p_{\mathrm{p}} = H_{\mathrm{R}} A_7 \overline{w}^2, \quad \text{if } |\chi| > A_6 \text{ and } \overline{w} \geq A_{18} \tag{5.3-70}$$

$$\Delta p_{\mathrm{p}} = -H_{\mathrm{R}} A_8 \overline{w}^2, \quad \text{if } |\chi| > A_6 \text{ and } \overline{w} < -A_{18} \tag{5.3-71}$$

$$\Delta p_{\mathrm{p}} = H_{\mathrm{R}} A_7 \overline{w} A_{18}, \quad \text{if } |\chi| > A_6 \text{ and } A_{18} > \overline{w} \geq 0 \tag{5.3-72}$$

$$\Delta p_{\mathrm{p}} = -H_{\mathrm{R}} A_8 \overline{w} A_{18}, \quad \text{if } |\chi| > A_6 \text{ and } 0 > \overline{w} \geq -A_{18} \tag{5.3-73}$$

where

Δp_{p} = the pump head

H_{R} = the reference pump head

A_1 ,… A_8 = user-supplied coefficients

For negative pump speeds, or flow reversal, the pump head is taken as

$$\Delta p_{\mathrm{p}} = \begin{cases} H_R A_7 d_1^2 & \text{if } d_1 \geq 0 \\ H_R A_8 d_1^2 & \text{if } d_1 < 0 \end{cases} \tag{5.3-74}$$

with

$$d_1 = \overline{w} - A_{10}\overline{s},$$

where

A_{10} = user-supplied coefficient

Pump Torque

Next, for positive pump speeds, the pump torque is

$$\overline{\tau} = \overline{s}^2 \left(A_{11} + A_{12}\chi + A_{13}\chi^2 + A_{14}\chi^3 + A_{15}\chi^4\right), \quad \text{if } \left|\frac{\overline{w}}{\overline{s}}\right| \leq A_6 \tag{5.3-75}$$

$$\overline{\tau} = A_{16}\overline{w}^2, \quad \text{if } \left|\frac{\overline{w}}{\overline{s}}\right| > A_6,\ \overline{w} \geq 0 \tag{5.3-76}$$

$$\overline{\tau} = A_{17}\overline{w}^2, \quad \text{if } \left|\frac{\overline{w}}{\overline{s}}\right| > A_6,\ \overline{w} < 0 \tag{5.3-77}$$

with

$$\overline{\tau} = \frac{\tau}{\tau_{\text{R}}} \tag{5.3-78}$$

where

τ = the pump torque

τ_{R} = the reference pump torque

χ, $\overline{w}$, $\overline{s}$ = the same as for the pump head

A_{11}, … A_{15} = user-supplied coefficient

For negative pump speeds, the pump torque is taken as

$$\overline{\tau} = A_{16}\left(\overline{w} - A_{20}\overline{s}\right)^2, \quad \text{if } \overline{w} \geq A_{20}\overline{s} \tag{5.3-79}$$

$$\overline{\tau} = A_{17}\left(\overline{w} - A_{20}\overline{s}\right)^2, \quad \text{if } \overline{w} < A_{20}\overline{s} \tag{5.3-80}$$

Pump Speed

The pump speed and pump torque are related by the equation is

$$I\frac{\text{ds}}{\text{dt}} = \tau_{\text{m}} - \tau_{\text{p}} - k_1 w \tag{5.3-81}$$

where

I = the moment of inertia of the pump and motor

s = the pump speed

τ_{m} = the motor torque

τ_{p} = the pump torque

k_1 = constant of proportionality

w = the pump flow rate

The term $-k_1$ w is a drag term proportional to the flow rate. The motor torque τ_{m} is a user-supplied table look-up.

Rotor Lock-Up

If the normalized pump speed, $\overline{s}$, falls below the value A_9, and if $A_9 > 0$, then the rotor locks; and the speed is held at zero after that.

Flow Rate Coefficients

The contribution to the a's in Eq. (5.2-11), Eq. (5.2-13), Eq. (5.2-15) and Eq. (5.2-17) are then as follows:

$$\Delta a_0 = \Delta a_0^{\text{pipe}} \tag{5.3-82}$$

$$\Delta a_1 = \Delta p_{\text{p}} + \Delta a_1^{\text{pipe}} \tag{5.3-83}$$

$$\begin{aligned}\Delta a_2 &= (\Delta t)^2 \frac{\partial \Delta \mathrm{p_p}}{\partial \mathrm{t}} + \Delta a_2^{\mathrm{pipe}} \\ &= (\Delta t)^2 \frac{\partial \Delta \mathrm{p_p}}{\partial \mathrm{s}} \frac{\partial \mathrm{s}}{\partial \mathrm{t}} + \Delta a_2^{\mathrm{pipe}}\end{aligned} \tag{5.3-84}$$

$$\Delta a_3 = \Delta t \frac{\partial}{\partial \mathrm{w}} \Delta p_{\mathrm{p}} + \Delta a_3^{\mathrm{pipe}} \tag{5.3-85}$$

Δp_{p} = the pump head

$\frac{\partial \Delta \mathrm{p_p}}{\partial \mathrm{t}}$, $\frac{\partial \mathrm{s}}{\partial \mathrm{t}} \frac{\partial \Delta \mathrm{p_p}}{\partial \mathrm{w}}$ = the partial derivatives of pump head and pump speed described above under the various conditions of speed and flow

5.3.4.2.2 Option 2

This pump option makes use of the homologous pump curves and permits operation of the pump in all four quandrants of pump operations – normal pump mode, energy dissipation mode, turbine mode, and reverse pump mode. In homologous pump theory, the homologous curves depict the pump performance map which relates the pump output variables to its input variables. The homologous curves are valid for a class of pumps that are geometrically and dynamically similar, independent of the fluid. The basis is that the physical parameters that describe the pump geometry, the fluid conditions, and the operating performance are functionally related. Therefore performance data acquired from a test or prototype pump can then be applied to predict the performance of other "similar" pumps. Default curves used for the pump head and torque are built into the model using data statements and are the three-region curve fit taken from a memo by J. F. Koenig for a pump with a specific speed (N_s) of 1800 rpm [5-1]. Users may override the defaults by defining a positive value for **IPMDFT**, defining polynomial coefficients via **PMPHD** and **PMPTQ**, and then selecting those coefficients by defining *APMPHD*(18).

The pump head and torque values are obtained from

$$H = H_{\mathrm{r}} \left(\overline{w}^2 + \overline{s}^2\right) \sum_{\mathrm{i}=0}^{6} C_{\mathrm{ji}} x^i \tag{5.3-86}$$

$$T = T_{\mathrm{r}} \left(\overline{w}^2 + \overline{s}^2\right) \sum_{\mathrm{i}=0}^{8} D_{\mathrm{ji}} x^i \tag{5.3-87}$$

where

H = pump head

T = pump torque

w = flow rate

s = pump speed

$x = \pi + \left(\frac{\overline{w}}{\overline{s}}\right) = \pi + \mathrm{atan2}\left(\overline{w}, \overline{s}\right)$

and the bar denotes normalization with respect to reference quantities(i.e., $\overline{w} = w/w_{\mathrm{r}}$) and the subscript r denotes reference quantities. The coefficients C_{ji} and D_{ji} represent the three-region curve fit to a sixth-order polynomial with:

$$j = \begin{cases} 1 & 0 \le x \le 3.14159 \\ 2 & 3.14159 < x \le 4.7124 \\ 3 & 4.7124 < x \end{cases}$$

The value for H_{r} is an input while T_{r} is obtained from

$$T_{\mathrm{r}} = \frac{H_{\mathrm{r}} w_{\mathrm{r}}}{(s_{\mathrm{r}} \eta \rho)} \tag{5.3-88}$$

where η is the pump efficiency at reference conditions and ρ is the fluid density.

The coefficients C_{ji} and D_{ji} are provided in Table 5.3.2 and Table 5.3.3.

Table 5.3.2: Default pump head coefficients for option 2 of the centrifugal pump model.

i	j=1	j=2	j=3
0	6.3380980E-01	4.3196699E+02	6.1719821E+03
1	4.6015764E-01	-5.7661438E+02	-4.9589692E+03
2	-2.4004049E+00	3.0100029E+02	1.4063329E+03
3	3.1793724E+00	-7.5465856E+01	-1.2617344E+02
4	-1.7730449E+00	8.6754986E+00	-1.3217121E+01
5	4.6235776E-01	-2.6062352E-01	3.2450530E+00
6	-4.6246398E-02	-1.5962870E-02	-1.6925040E-01

Table 5.3.3: Default pump torque coefficients for option 2 of the centrifugal pump model.

i	j=1	j=2	j=3
0	-6.8436766E-01	-1.1549471E+03	-3.7981080E+02
1	2.7759909E+00	1.8584915E+03	7.2614914E+02
2	-5.3988010E+00	-1.2376683E+03	-4.9625029E+02
3	6.8541205E+00	4.3601653E+02	1.6764136E+02
4	-4.0757860E+00	-8.5573772E+01	-3.0366923E+01
5	1.0813311E+00	8.8627717E+00	2.8311896E+00
6	-1.0475812E-01	-3.7830487E-01	-1.0681625E-01
7	0.0	0.0	0.0
8	0.0	0.0	0.0

At low flow conditions with $|\overline{w}| < w_{\mathrm{c}}$ where w_{c} represents a cutoff value specified by the user, the pump head is calculated as a linear interpolation between the pump head at zero flow and at the w_{c} flow:

$$H(w, s) = \frac{\left(H\left(w_{\mathrm{c}}, \overline{s}\right) \overline{w} + H\left(0, \overline{s}\right)\left(w_{\mathrm{c}} - \overline{w}\right)\right)}{w_{\mathrm{c}}} \tag{5.3-89}$$

where

$$w_{\mathrm{c}} = A_{13} \frac{\overline{w}}{|\overline{w}|} \tag{5.3-90}$$

A_{13} is specified as an input value in APMPHD (13,IPMP) in input block 18. The frictional torque losses, T_{f}, are taken as a three-region quadratic fit over $\overline{s}$

$$T_{\mathrm{f}} = T_{\mathrm{r}}\left(A_{\mathrm{j}1} + A_{\mathrm{j}2}\overline{s} + A_{\mathrm{j}3}\overline{s}\,|\overline{s}|\right) \tag{5.3-91}$$

where

$$j = \begin{cases} 1 & \overline{s} < s_1 \\ 2 & s_1 < \overline{s} < s_2 \\ 3 & s_2 < \overline{s} \end{cases}$$

The nine coefficients, A_{ji}, are specified as input values in *APMPHD*. The cutoff speeds s_1 and s_2 are also input values in *APMPHD*. The recommended values for the nine coefficients and cutoff speeds are presented in Table 5.3.4.

Table 5.3.4: Recommended values for the torque loss coefficients in option 2 for the centrifugal pump.

k	Variable	APMPHD(K,IPMP)
1	$A_{1,1}$	0.01
2	$A_{2,1}$	0.0
3	$A_{3,1}$	-73.13
4	$A_{1,2}$	0.00268
5	$A_{2,2}$	0.0
6	$A_{3,2}$	0.07
7	$A_{1,3}$	0.00383
8	$A_{2,3}$	0.01071
9	$A_{3,3}$	0.01406
10	s_1	0.01
11	s_2	0.268

For the steady-state, the subroutine SSPUMP makes use of a root finder to locate the pump speed necessary for a specified pump head. It is assumed that steady-state will be at a flow level above w_{c}.

The subroutine PUMPFL includes the following calculations for Δa_{o}, Δa_1, Δa_2 and Δa_3, with

$$\Delta a_0 = \Delta a_0^{\text{pipe}} \tag{5.3-92}$$

$$\Delta a_1 = H + \Delta a_1^{\text{pipe}} \tag{5.3-93}$$

$$\Delta a_2 = \Delta t^2 \frac{\partial}{\partial \text{t}} H + \Delta a_2^{\text{pipe}} \tag{5.3-94}$$

$$\Delta a_3 = \Delta t \frac{\partial}{\partial \text{w}} H + \Delta a_3^{\text{pipe}} \tag{5.3-95}$$

where H and Δt represent the pump head and time interval, respectively. The terms Δa_{o} and Δa_1 are calculated as in the centrifugal pump option 1; Δa_2 is found using:

$$\frac{\partial}{\partial \text{t}} H = \frac{\partial \overline{s}}{\partial \text{t}} \frac{\partial \text{H}}{\partial \text{s}} \tag{5.3-96}$$

where

$$\frac{\partial \overline{\text{s}}}{\partial \text{t}} = \frac{(T_{\text{m}} - T_{\text{p}} - T_{\text{f}})}{(s_{\text{r}} I)} \tag{5.3-97}$$

T_{m} = motor torque,

I = pump inertia

and from Eq. (5.3-85) and Eq. (5.3-88)

$$\frac{\partial \text{H}}{\partial \overline{\text{s}}} = 2 H_{\text{r}} \overline{s} \sum_{\text{i}=0}^{6} c_{\text{ji}} x^i - H_{\text{r}} \overline{w} \sum_{\text{i}=1}^{6} c_{\text{ji}} i x^{i-1} \tag{5.3-98}$$

for $|\overline{w}| > w_c$and

$$\frac{\partial \text{H}}{\partial \overline{\text{s}}} = \frac{\left[\frac{\partial \text{H}}{\partial \overline{\text{s}}} (w_{\text{c}}, \overline{s}) \, \overline{w} + \frac{\partial \text{H}(0, \overline{s})}{\partial \overline{\text{s}}} (w_{\text{c}} - \overline{w}) \right]}{w_{\text{c}}} \tag{5.3-99}$$

for $|\overline{w}| < w_{\mathrm{c}}$. T_{m} is defined using *AMOTTK* or using JTYPE=-2 within the control system (see Table 2.7.6).

Δa_3 is found using Eq. (5.3-95) and

$$\begin{aligned}\frac{\partial \mathrm{H}}{\partial \overline{w}} &= 2H_{\mathrm{r}}\overline{w}\sum_{\mathrm{i}=0}^{6} c_{\mathrm{ji}}x^{i} + H_{\mathrm{r}}\overline{s}\sum_{\mathrm{i}=1}^{6} C_{\mathrm{ji}}ix^{i-1} \\ \frac{\partial \mathrm{H}}{\partial \mathrm{w}} &= \frac{1}{w_{\mathrm{r}}}\frac{\partial \mathrm{H}}{\partial \overline{\mathrm{w}}}\end{aligned} \tag{5.3-100}$$

for $|\overline{w}| > w_{\mathrm{c}}$ and from

$$\frac{\partial \mathrm{H}}{\partial \overline{\mathrm{w}}} = \frac{\left(H\left(w_{\mathrm{c}}, \overline{s}\right) - H\left(0, \overline{s}\right)\right)}{w_{\mathrm{c}}} \tag{5.3-101}$$

for $|\overline{w}| < w_{\mathrm{c}}$, and

The subroutine PUMPFN includes the pump torque curves in integrating Eq. (5.3-97) to determine s. Approximating Eq. (5.3-97) as

$$I\frac{\Delta s}{\Delta t} = T_{\mathrm{m}} - \frac{1}{2}\left(T_3 + T_4\right) - \frac{1}{2}\left(T_{\mathrm{f3}} + T_{\mathrm{f4}}\right) \tag{5.3-102}$$

Note where the subscripts 3 and 4 denote the beginning and ending times of the Δt interval and approximating

$$T_4 = T_3 + \frac{\partial \mathrm{T}}{\partial \mathrm{s}}\Delta s + \frac{\partial \mathrm{T}}{\partial \mathrm{w}}\Delta w \tag{5.3-103}$$

$$T_{\mathrm{f4}} = T_{\mathrm{f3}} + \frac{\partial \mathrm{T_f}}{\partial \mathrm{s}}\Delta s \tag{5.3-104}$$

in Eq. (5.3-102), Δs may be obtained as

$$\Delta s = \frac{T_{\mathrm{m}} - T_3 - \frac{1}{2}\frac{\partial \mathrm{T}}{\partial \mathrm{w}}\Delta w - T_{f3}}{\left[\frac{I}{\Delta t} + \frac{1}{2}\frac{\partial \mathrm{T}}{\partial \mathrm{s}} + \frac{1}{2}\frac{\partial T_{\mathrm{f}}}{\partial \mathrm{s}}\right]} \tag{5.3-105}$$

Locked Rotor

If $\overline{s} < A_{20}$, where A_{20} is the input variable APMPHD(20, IPMP), and if $\overline{w} < A_{19}$, then the rotor locks and the pump speed is set to zero and held at zero. For a locked rotor the pump head is calculated as

$$H = \begin{cases} -H_{\mathrm{r}}A_{12|14}\left|\overline{w}\right|\overline{w} & \text{if } |\overline{w}| \geq A_{13} \\ -H_{\mathrm{r}}A_{12|14}A_{13}\overline{w} & \text{if } |\overline{w}| < A_{13} \end{cases} \tag{5.3-106}$$

$$A_{12|14} = \begin{cases} A_{12} & \text{if } \overline{w} \geq 0.0 \\ A_{14} & \text{if } \overline{w} < 0.0 \end{cases} \tag{5.3-107}$$

where A_{12}, A_{13}, and A_{14} are input in the APMPHD(K, IPMP) array.

Pump Cavitation

In a centrifugal pump, cavitation will occur if the applied net positive suction head (ANPSH), or pump inlet pressure, is less than the required net positive suction head (RNPSH). This can occur if there is a break in the outlet pipe from the pump, leading to an increase in the pump flow rate and an increase in the RNPSH. If pump cavitation occurs, there will be a drop in pump head. Although there is reasonably good RNPSH data for many pumps, there is little data on pump behavior in the cavitation range, mainly because cavitation tends to destroy a pump, so no one wants to operate in that range. On the other hand, transient pump cavitation can occur for a short time after a pipe rupture, and the resulting decrease in pump head can have a significant impact on the

flow out the rupture and on the flow rate through the core. The homologous pump model in PRIMAR-4 tests for pump cavitation and reduces the pump head if cavitation occurs.

The required net positive suction head is calculated as

$$\text{RNPSH} = \bar{s}^2 \left(C_{\text{c}0} + C_{\text{c}1} Z + C_{\text{c}2} Z^2 + C_{\text{c}3} Z^3 + C_{\text{c}4} Z^4 \right) \tag{5.3-108}$$

where

$$Z = \frac{\overline{w}}{\bar{s}} \tag{5.3-109}$$

Then, if RNPSH > ANPSH, the cavitated head, H_{c}, is calculated as

$$H_{\text{c}} = f_{\text{c}} H \tag{5.3-110}$$

where H is the normal head, calculated as described in the previous sections, and the cavitation factor f_{c} is calculated as

$$f_{\text{c}} = \frac{1}{1 + 1000 x^2} \tag{5.3-111}$$

where

$$x = \frac{\text{RNPSH} - \text{ANPSH}}{\text{ANPSH}} \tag{5.3-112}$$

The cavitation factor of Eq. (5.3-111) is somewhat arbitrary; but, as previously mentioned, there is little data for behavior in the cavitation range, and this expression results in a rapid drop in pump head as the pump goes farther into the cavitation range. With this treatment, the expressions used in Eq. (5.3-95) and Eq. (5.3-96) for Δa_2 and Δa_3 are

$$\frac{\partial \text{H}_{\text{c}}}{\partial \overline{\text{s}}} = f_{\text{c}} \left[\frac{\partial \text{H}}{\partial \overline{\text{s}}} - 2000 x H \frac{\partial \text{x}}{\partial \overline{\text{s}}} \right] \tag{5.3-113}$$

$$\frac{\partial \text{H}_{\text{c}}}{\partial \overline{\text{w}}} = f_{\text{c}} \left[\frac{\partial \text{H}}{\partial \overline{\text{w}}} - 2000 x H \frac{\partial \text{x}}{\partial \overline{\text{w}}} \right] \tag{5.3-114}$$

5.3.4.2.3 EBR-II Pump Model

The third centrifugal pump option in PRIMAR-4 is essentially the pump model used in the NATDEMO code [5-5] for EBR-II. By adjusting the coefficients in the pump characteristics curves, this model could probably also be used for other reactors. This model is somewhat simpler than the option 2 model, in that the EBR-II pump model is mainly applicable to the first quadrant (positive flow, positive pump speed). This model can be used for negative flow, but it is probably not applicable to negative pump speed. Also, for the EBR-II pump model, the user specifies the pump speed as a function of time; whereas in options 1 and 2, the code calculated the pump speed.

In this option, the pump head is calculated from

$$\overline{H} = b_1 \bar{s}^2 + b_2 \overline{sw} + b_3 \frac{\overline{w}}{|\overline{w}|} |\overline{w}|^{b_4} \tag{5.3-115}$$

where

$$\overline{H} = H / H_{\text{r}} \tag{5.3-116}$$

$$\overline{s} = s/s_{\mathrm{r}} \tag{5.3-117}$$

$$\overline{w} = w/w_{\mathrm{r}} \tag{5.3-118}$$

H = pump head

s = pump speed

and

w = coolant flow rate

The r subscript refers to the rated value. The coefficients b_1 and b_2 are constant, but the values of b_3 and b_4 depend on the speed and flow rate:

$$b_3 = b_{3\mathrm{m}} \quad \text{if } \overline{s} > \varepsilon_{\mathrm{m}}\overline{w} \tag{5.3-119}$$

$$b_3 = b_{3\mathrm{t}} \quad \text{if } \overline{s} \leq \varepsilon_{\mathrm{m}}\overline{w} \text{ and } w \geq \overline{w}_{\mathrm{t}} \tag{5.3-120}$$

$$b_3 = b_{3\mathrm{l}} \quad \text{if } \overline{s} \leq \varepsilon_{\mathrm{m}}\overline{w} \text{ and } w < \overline{w}_{\mathrm{t}} \tag{5.3-121}$$

$$b_4 = b_{4\mathrm{t}} \quad \text{if } \overline{s} > \varepsilon_{\mathrm{m}}\overline{w} \text{ or } w \geq \overline{w}_{\mathrm{t}} \tag{5.3-122}$$

$$b_4 = b_{4\mathrm{l}} \quad \text{if } \overline{s} > \varepsilon_{\mathrm{m}}\overline{w} \text{ or } w < \overline{w}_{\mathrm{t}} \tag{5.3-123}$$

Eq. (5.3-119) and Eq. (5.3-122) are based on the idea of using a stopped-rotor pressure drop if $\overline{s} < \varepsilon_{\mathrm{m}}\overline{w}$. Also, the stopped-rotor pressure drop is a laminar value if $\overline{w} < \overline{w}_{\mathrm{t}}$, or a turbulent value if $\overline{w} \geq \overline{w}_{\mathrm{t}}$. The default values for the coefficients are:

$$b_1 = 1.1740 \tag{5.3-124}$$

$$b_2 = .0818 \tag{5.3-125}$$

$$b_{3\mathrm{m}} = -.2558 \tag{5.3-126}$$

$$b_{3\mathrm{t}} = -.5923 \tag{5.3-127}$$

$$b_{3\mathrm{l}} = -.0471 \tag{5.3-128}$$

$$b_{4\mathrm{t}} = 1.9 \tag{5.3-129}$$

$$b_{4\mathrm{l}} = 1.0 \tag{5.3-130}$$

$$\overline{w}_{\mathrm{t}} = .06 \tag{5.3-131}$$

$$\varepsilon_{\mathrm{m}} = .55 \tag{5.3-132}$$

$$H_{\mathrm{r}} = 358,530 \text{ (Pa)} \tag{5.3-133}$$

$$w_{\mathrm{r}} = 250.2 \text{ (kg/s)} \tag{5.3-134}$$

and

$$s_r = 14.5 \text{ revolutions/s} = 870 \text{ rpm} \tag{5.3-135}$$

This correlation produces a discontinuity in pump head at the switch from spinning rotor to stopped rotor. With the default values, the pump head is continuous at the transition from turbulent to laminar flow.

For the steady-state initialization in SSPUMP, the initial normalized pump head, $\overline{H_0}$ and flow rate, $\overline{w_0}$, are known, and Eq. (5.3-115) is solved to find $\overline{s_0}$, the initial steady-state pump speed:

$$\overline{s}_{\mathrm{o}} = \frac{-b_2\overline{w}_0 + \left[b_2^2\overline{w}_0^2 - 4b_1\left(b_3\overline{w}_0^{b_4} - \overline{H}_0\right)\right]^{1/2}}{2b_1} \tag{5.3-136}$$

It is assumed that $\overline{s}_0 > \varepsilon_{\mathrm{m}}\overline{w}_0$, and the values used for b_3 and b_4 in SSPUMP are $b_{3\mathrm{m}}b_{4\mathrm{t}}$. The user supplies a table of normalized pump speed vs. time, normalized to the initial steady-state speed. The code then interpolates linearly from this table to get pump speeds for the transient calculation.

In PUMPFL where Δa_0, Δa_1, Δa_2, and Δa_3 are calculated, Δa_0 and Δa_1, are calculated using Eq. (5.3-82) and Eq. (5.3-83) or Eq. (5.3-92) and Eq. (5.3-93), as in the other pump options. For this option

$$\Delta a_2 = \Delta t^2 \frac{\mathrm{dH}}{\mathrm{ds}}\frac{\mathrm{ds}}{\mathrm{dt}} + \Delta a_2^{\mathrm{pipe}} \tag{5.3-137}$$

with

$$\frac{\mathrm{dH}}{\mathrm{ds}} = \frac{\left(2b_1\overline{s} + b_2\overline{w}\right)H_{\mathrm{r}}}{S_{\mathrm{r}}} \tag{5.3-138}$$

and ds/dt obtained from the user-specified table of normalized pump speed vs. time:

$$\frac{\mathrm{ds}}{\mathrm{dt}} = \frac{s\left(t+\Delta t\right) - s\left(t\right)}{\Delta t} \tag{5.3-139}$$

also,

$$\Delta a_3 = \Delta t\frac{\mathrm{dH}}{\mathrm{dw}} + \Delta a_3^{\mathrm{pipe}} = \Delta t\ H_{\mathrm{r}}\left[b_2\overline{s} + b_3b_4\left(\frac{\overline{w}}{|w|}\right)|w|^{b_4-1}\right] + \Delta a_3^{\mathrm{pipe}} \tag{5.3-140}$$

5.3.4.2.4 Pump Head vs. Flow Option

One pump option in SAS4A/SASSYS-1 allows the user to specify the normalized pump head as a function only of normalized flow rate. In this option, the normalization is to the initial steady-state values. The user supplies a table of normalized pump head vs. normalized flow rate, and the code uses linear interpolation between the entries in the table.

For this option, Δa_0 and Δa_1 for the pump are calculated in the same way as for the other pump options. Then

$$\Delta a_2 = 0 + \Delta a_2^{\mathrm{pipe}} \tag{5.3-141}$$

and

$$\Delta a_3 = \Delta t\frac{\mathrm{dH}}{\mathrm{dW}} + \Delta a_3^{\mathrm{pipe}} \tag{5.3-142}$$

with dH/dw being evaluated numerically from the table of $\overline{H}$ vs. $\overline{w}$:

$$\frac{\mathrm{dH}}{\mathrm{dw}} = \frac{H\left(w + .01w_0\right) - H\left(w\right)}{.01w_0} \tag{5.3-143}$$

where w_0 is the initial steady state flow rate, w is the current flow rate, and H is the pump head.

5.3.4.3 Electromagnetic Pumps

5.3.4.3.1 Electromagnetic Pump

The electromagnetic pump is modeled as a pipe element. The expression for the pump head for the electromagnetic pump is

$$\Delta p_{\mathrm{p}} = H_{\mathrm{s}}(t)\left(1 - \frac{v_{\mathrm{Na}}}{v_{\mathrm{sy}}}\right) \tag{5.3-144}$$

with

$$v_{\mathrm{Na}} = \frac{w}{\rho A} \tag{5.3-145}$$

where

Δp_{p} = the pump head

$H_{\mathrm{s}}(t)$ = the pump stall head

v_{Na} = the sodium velocity at the beginning of the time interval

w = the pipe flow rate

A = the pipe area

ρ = the sodium density

v_{sy} = user-supplied synchronous magnetic field velocity

The quantity

$$\frac{H_{\mathrm{s}}(t)}{H_{\mathrm{s}}(o)} \tag{5.3-146}$$

is a user-supplied table of normalized stall head as a function of time.

The contributions to the a's in Eq. (5.2-11) through Eq. (5.2-17) for the electromagnetic pump are

$$\Delta a_0 = \Delta a_0^{\mathrm{pipe}} \tag{5.3-147}$$

$$\Delta a_1 = \Delta t \Delta p_{\mathrm{p}}(t_3) + \Delta a_1^{\mathrm{pipe}} \tag{5.3-148}$$

$$\Delta a_2 = \Delta t\left[H_{\mathrm{s}}(t_4) - H_{\mathrm{s}}(t_3)\right]\left(1 - \frac{v_{\mathrm{Na}}}{v_{\mathrm{sy}}}\right) + \Delta a_2^{\mathrm{pipe}} \tag{5.3-149}$$

$$\Delta a_3 = -\Delta t H_{\mathrm{s}}(t_3)\, v_{\mathrm{Na}} + \Delta a_3^{\mathrm{pipe}} \tag{5.3-150}$$

where

$\Delta p_{\mathrm{p}}(t_3)$ = the pump head evaluated at the beginning of the time interval

Δt = the time interval

$H_{\mathrm{s}}(t_3)$ = the stall head at the beginning of the time interval

$H_{\mathrm{s}}(t_4)$ = the stall head at the end of the time interval

5.3.4.3.2 ALMR EM Pumps

In the ALMR design, the primary pumps contain motor-generators that extend the coastdown times of these EM pumps. Since this combination of EM pumps and motor-generators cannot be modeled well with previous SAS4A/SASSYS-1 pump options, the ALMR program developed a new pump model to handle this configuration. The new model uses a model similar to that used by General Electric in their ARIES-P code. It is based on head and efficiency data obtained by GE and transmitted to ANL (5-29).

In the ALMR design, a synchronous motor is running all of the time during normal operation, but the power to the EM pump does not go through the motor as long as the normal pump power is available. If normal pump power is lost, then the motor becomes a generator, and a switch is thrown automatically to supply voltage from the motor-generator to the pump. The pump coastdown rate is then determined by the inertia of the motor. The motor is designed such that it will initially supply 60% of nominal voltage to the pump. Thus, when normal power is lost and the motor-generator power is switched on there is a sudden drop in pump head and flow followed by a gradual coastdown.

Controlling the Trip Time

There are two options for controlling the time at which the ALMR EM Pump cuts over to the motor-generator. The first method is to set the trip time in input variable APMPHD(7,IPMP) in block PMR4IN. The second method is to set a dynamic trip time using a function. In this case, the *Function Block* ID is set in input variable **ILRPMP**. A single argument is passed to the function containing the value of the time at the end of the current PRIMAR-4 time step. If the time exceeds the value returned by the function, then pump cuts-over to the motor generators, and remains in a tripped state for the remainder of the simulation.

Pump Head

The pump head is correlated with an expression of the form

$$\overline{H} = \left(\frac{\overline{V}}{\overline{f}}\right)^{3.5} h_{\mathrm{n}}\left(\frac{\overline{w}}{\overline{f}}\right) - L_{\mathrm{f}}\overline{w}^2 \tag{5.3-151}$$

where the normalized terms are defined as

$$\overline{H} = \frac{H}{H_{\mathrm{r}}} \tag{5.3-152}$$

$$\overline{V} = \frac{V}{V_{\mathrm{r}}} \tag{5.3-153}$$

$$\overline{f} = \frac{f}{f_{\mathrm{r}}} \tag{5.3-154}$$

$$\overline{w} = \frac{w}{w_{\mathrm{r}}} \tag{5.3-155}$$

where

H = pump head

H_{r} = rated head

V = pump voltage

V_{r} = rated voltage

f = frequency

f_{r} = rated frequency

w = mass flow rate

w_{r} = rated mass flow rate

L_{f} = friction loss coefficient

and h_{n} is a head curve correlated as

$$h_{\mathrm{n}}\left(\frac{\overline{w}}{\overline{f}}\right)=\sum_{\mathrm{j}=1}^{5}a_{\mathrm{j}}\left(\frac{\overline{w}}{\overline{f}}\right)^{j-1} \tag{5.3-156}$$

with the coefficients a_{j} determined by a least-squares fit to the data.

Pump Efficiency

The pump efficiency, ε_{f}, is correlated as

$$\varepsilon_{\mathrm{f}}=F\left(\overline{V}\right)G\left(\frac{\overline{w}}{\overline{f}}\right)\varepsilon_{\mathrm{fr}} \tag{5.3-157}$$

with

$$F\left(\overline{V}\right)=\sum_{\mathrm{j}=1}^{7}b_{\mathrm{j}}\overline{V}^{j-1} \tag{5.3-158}$$

$$G\left(\frac{\overline{w}}{\overline{f}}\right)=\begin{cases}0.01 & \text{if } \frac{\overline{w}}{\overline{f}}\geq 5\\ \sum_{\mathrm{j}=1}^{9}c_{\mathrm{j}}\left(\frac{\overline{w}}{\overline{f}}\right)^{j-1} & \text{if } \frac{\overline{w}}{\overline{f}}\leq 5\end{cases} \tag{5.3-159}$$

and $\varepsilon_{\mathrm{fr}}$ is the rated efficiency

Pump Voltage

Before the cut-over to the motor-generator, the pump voltage is assumed to be constant at its rated value. Also, the frequency is constant at its rated value. Immediately after cut-over, the voltage drops to a fraction, V_{fr}, of its initial value. Then the voltage is proportional to the square of the frequency, so after cut-over the voltage is

$$\overline{V}=V_{\mathrm{fr}}\overline{f}^{2} \tag{5.3-160}$$

Motor Speed

The equation for the motor speed, s, is

$$\frac{\mathrm{ds}}{\mathrm{dt}}=-\frac{\tau_{\mathrm{p}}+\tau_{\mathrm{l}}}{I} \tag{5.3-161}$$

where

τ_{p} = pump torque

τ_{l} = friction loss

I = moment of inertia

Note that the motor speed and the pump frequency are the same:

$$f=s \tag{5.3-162}$$

The pump torque is given by

$$\tau_{\mathrm{p}}=\frac{Hw}{\varepsilon_{\mathrm{f}}\rho s} \tag{5.3-163}$$

where

ρ = liquid density

The friction loss in the motor is assumed to have the form

$$\tau_{\mathrm{l}} = \tau_{\mathrm{r}} L_{\mathrm{m}} s \tag{5.3-164}$$

where L_{m} is a loss coefficient and τ_{r}, the rated torque, is given by Eq. (5.3-163) with rated values used for all terms.

Correlations to Pump Data

Reference 5-29 transmitted the data taken for the pump. This data has been fit by a least-squares fitting program to give the parameters listed in Table 5.3.5 for use in Eq. (5.3-156), Eq. (5.3-158), and Eq. (5.3-159). Also a value of .07592 is used for L_{f} in Eq. (5.3-151). Table 5.3.6 lists the data from Ref. 5-29, as well as the head and efficiency values calculated using the correlations of equations Eq. (5.3-151), Eq. (5.3-156), Eq. (5.3-157), and Eq. (5.3-159) with the coefficients in Table 5.3.5. Also, some of this data is plotted in Figure 5.3.1 and Figure 5.3.2. It can be seen that the correlations match the data well except for a couple of points.

Table 5.3.5: Correlation Coefficients for Use with the ALMR EM Pump Model

j	a_{j}	b_{j}	c_{j}
1	1.133	-0.148	0.0
2	0.996	7.110	-51.235
3	-2.498	-15.972	684.934
4	6.056	9.942	-3483.628
5	-4.611	12.024	9119.690
6	-	-18.536	-13449.761
7	-	6.577	11279.948
8	-	-	-5014.503
9	-	-	915.555

Table 5.3.6: Pump Head and Efficiency for the ALMR EM Pump Model

Normalized Voltage	Normalized Frequency	Normalized Flow	Measured Normalized Head	Fit Normalized Head	Measured Efficiency	Fit Efficiency
1.000	1.000	1.148	0.042	0.041	0.115	0.133
1.000	1.000	1.124	0.250	0.242	0.505	0.469
1.000	1.000	1.086	0.500	0.524	0.781	0.807
1.000	1.000	1.000	1.000	1.000	1.000	0.997
1.000	1.000	0.800	1.483	1.494	0.928	0.915
1.000	1.000	0.600	1.508	1.514	0.710	0.709
1.000	1.000	0.400	1.399	1.389	0.469	0.468
1.000	1.000	0.200	1.265	1.270	0.229	0.229
1.000	1.000	0.000	1.134	1.133	0.000	0.000
0.627	0.778	0.886	0.042	0.042	0.183	0.256
0.627	0.778	0.862	0.167	0.158	0.551	0.600
0.627	0.778	0.824	0.333	0.314	0.817	0.879
0.627	0.778	0.762	0.581	0.500	0.970	0.932
0.628	0.778	0.762	0.581	0.505	0.970	0.932
0.312	0.472	0.532	0.042	0.052	0.318	0.381
0.312	0.472	0.520	0.083	0.097	0.529	0.608
0.312	0.472	0.495	0.167	0.178	0.773	0.848
0.312	0.472	0.476	0.227	0.226	0.863	0.884
1.202	1.111	1.143	1.306	1.144	1.008	1.009
0.929	0.944	0.952	0.905	0.912	0.998	1.003
0.820	0.833	0.857	0.734	0.835	0.986	0.981
0.519	0.667	0.667	0.444	0.415	0.954	0.932
0.437	0.556	0.571	0.327	0.383	0.917	0.920
0.251	0.500	0.476	0.227	0.096	0.855	0.789
0.219	0.389	0.381	0.145	0.143	0.757	0.757
0.153	0.278	0.286	0.082	0.111	0.567	0.602
0.071	0.222	0.190	0.037	0.025	0.268	0.255
0.027	0.111	0.095	0.009	0.010	0.026	0.031

5.3.4.3.3 Equivalent Circuit EM Pump

Equivalent circuit theory enables generalized systems-level modeling of EM pumps based on an electric circuit with components that represent the main responses of the pump. By representing the various dissipative and inertial aspects of the pump with resistors and inductors within a hypothetical electrical circuit, traditional circuit analysis techniques are used to approximate the pump's internal energy flow. Benefits of equivalent circuit theory within SAS include generic pump control as functions of voltage and frequency, flexibility to model a range of pump designs and operating conditions, and computational efficiency.

Two pump models are available on the basis of equivalent circuit theory, referred to as the "*detailed*" and "*simple*" models. With the exception of the governing circuit and consequent governing equations, the two models share the same framework, features, and calculation procedures. The common features are discussed here, while the details that differ between the two models are discussed in the subsections below and Section 5.16.7.

The equivalent circuit models assume that the pump is a three-phase, wye-connected, linear electromagnetic pump (annular or non-annular, depending on the specification of *ITYPEL*). Assuming each phase is equally loaded, all phases are represented using a single circuit of either detailed or simple type. Equating the circuit

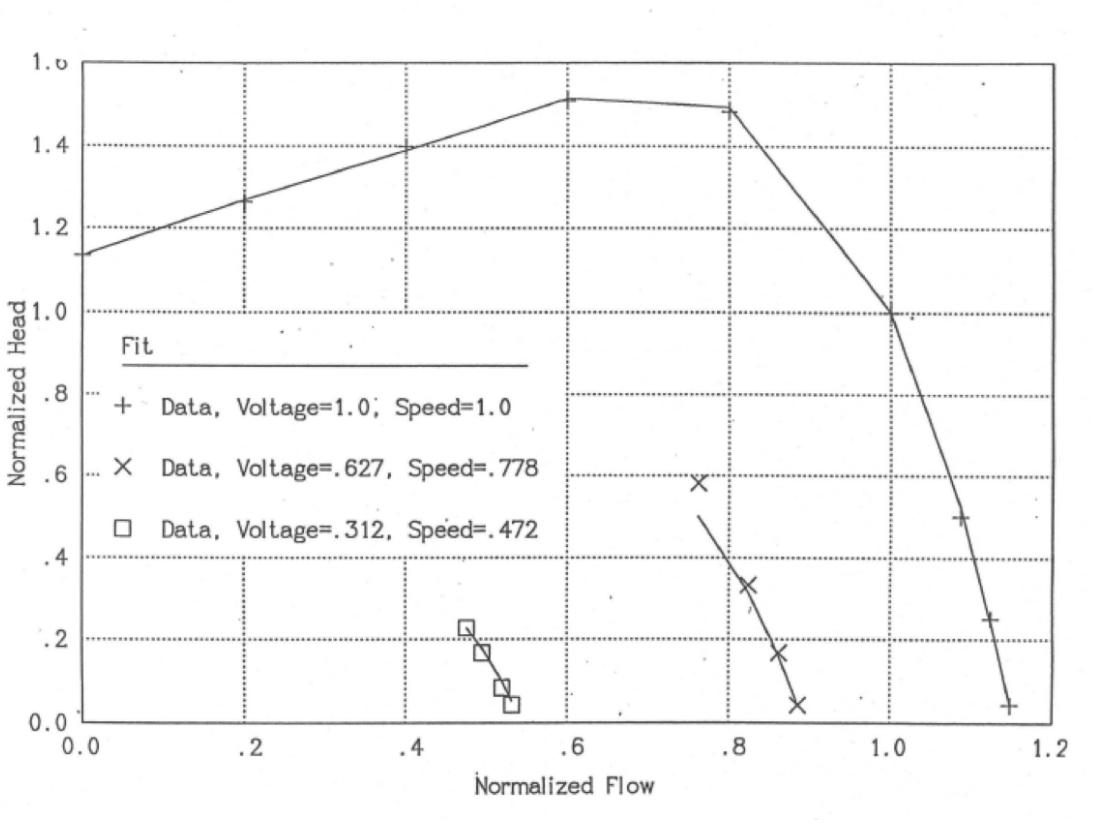


Figure 5.3.1: Pump Head, Data and Fits for the ALMR EM Pump Model.

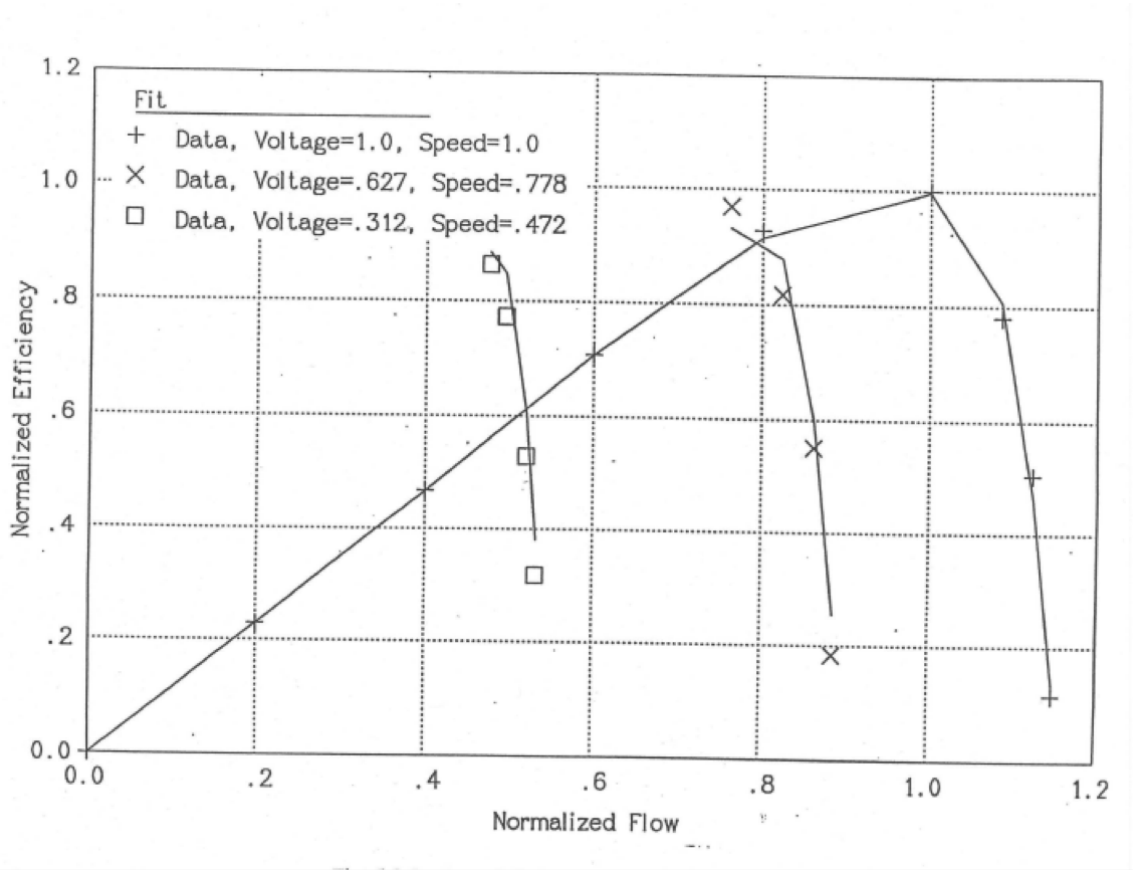


Figure 5.3.2: Pump Efficiency, Data and Fits for the ALMR EM Pump Model.

prediction for electrical power dissipation in resistors representing fluid mechanical power with the hydraulic power imparted to the fluid, the pressure developed by the pump (not including *friction, gravity, or form losses*) is predicted as

$$\Delta p_P = \frac{3I^2\Psi(s)}{Q} \tag{5.3-165}$$

where

Δp_P = pump developed pressure

I = total phase current

Q = fluid volumetric flowrate

$\Psi(s)$ = model-dependent circuit transform function

$s = 1 - \frac{v_f}{v_s}$ = slip

v_f = fluid velocity

v_s = pump synchronous speed

Due to the limited inertia of EM pumps, a quasi-static assumption is employed in solving for I and Ψ. Therefore the steady-state circuit equations are solved at each transient time step.

Based on this expression, the contributions to the $\Delta a_i(k)$'s in Eq. (5.2-11), Eq. (5.2-13), Eq. (5.2-15), and Eq. (5.2-17) are:

$$\Delta a_0(k) = \Delta a_0^{\text{pipe}} \tag{5.3-166}$$

$$\Delta a_1(k) = \Delta t \Delta p_P(k) + \Delta a_1^{\text{pipe}} \tag{5.3-167}$$

$$\Delta a_2(k) = \Delta a_2^{\text{pipe}} \tag{5.3-168}$$

$$\Delta a_3(k) = \Delta t \frac{\partial \Delta p_P(k)}{\partial w} + \Delta a_3^{\text{pipe}} \approx \Delta t \frac{\Delta p_P(k, 1.01(w + 10^{-6})) - \Delta p_P(k, w)}{0.01w + 1.01 * 10^{-6}} + \Delta a_3^{\text{pipe}} \tag{5.3-169}$$

Pump efficiency is calculated by dividing fluid power in the forward direction by the electrical power dissipated in the entire pump (including all three phases) as

$$\eta = \frac{(\Delta p_P - \Delta p_f) \cdot Q}{3V_s I \cos\phi} \tag{5.3-170}$$

where

η = efficiency

Δp_f = *friction losses in the pump element*

V_s = power source phase voltage

ϕ = phase angle

Arbitrary control of the pump's power supply is enabled through the use of *FUNCTION blocks* to individually control voltage and frequency with time.

The steady-state initialization process is outlined in Figure 5.3.3.

The resistance representing Joule heating is adjusted for fluid temperature using *TR*. In the case that *TR* is specified, the Joule resistance is assumed to be proportional to the temperature-dependent fluid electrical resistance

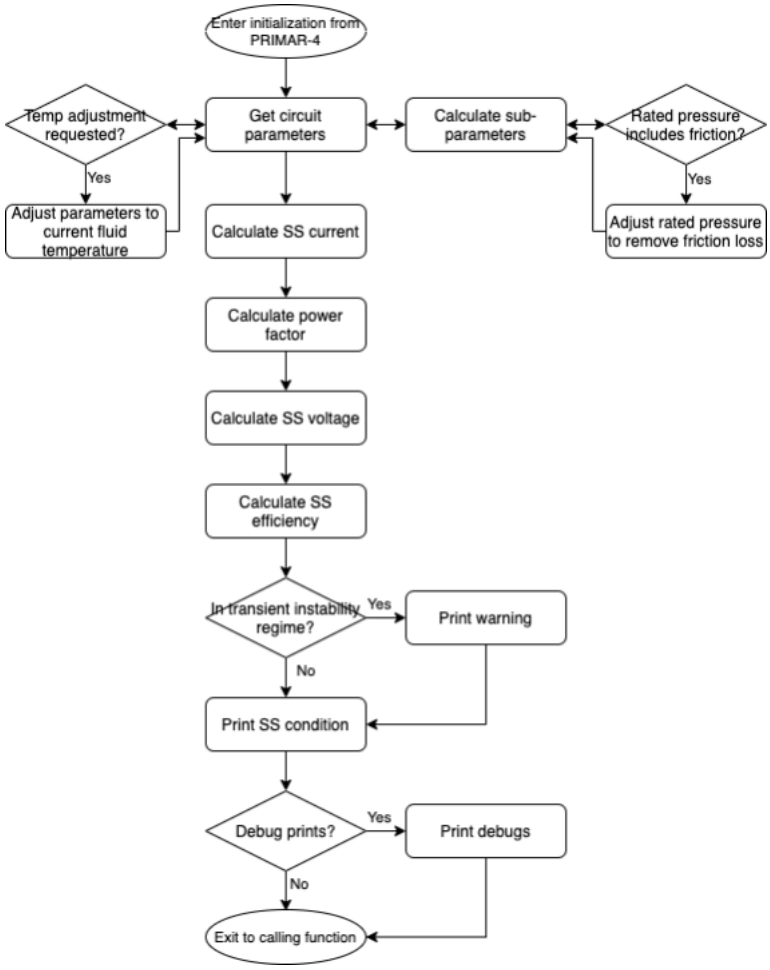


Figure 5.3.3: Flow chart of the steady-state initialization procedure for equivalent circuit pumps.

as discussed specifically for each model. The default correlation for sodium electrical resistivity is given in Eq. (5.3-171), with 371 K $\leq T \leq$ 1373 K *[5-10]*.

$$\rho_r(T) = C_1 + C_2 T + C_3 T^2 \tag{5.3-171}$$

where

C_1 = 7.756E-8 Ω-m

C_2 = 2.054E-10 Ω-m/C

C_3 = 3.481E-13 Ω-m/C^2

T = fluid temperature

The user may also enter their own correlation coefficients to be used in Eq. (5.3-171) as described in Table 5.3.8 for either sodium or when other coolants are employed (see *ICLPRP* and *INAS3D*).

Once circuit parameters are deduced, initialization uses the flow rate and pressure across the pump element as determined by PRIMAR-4 to determine the pump's current, power factor, voltage, and efficiency at steady-state. The steady-state pump voltage is then stored as the amplitude of the normalized voltage used during transient steps.

Pump heating may be modeled for equivalent circuit pumps using the scaling factor *fH*. This feature is discussed within the *temperature calculation section*.

5.3.4.3.3.1 Detailed Circuit Model

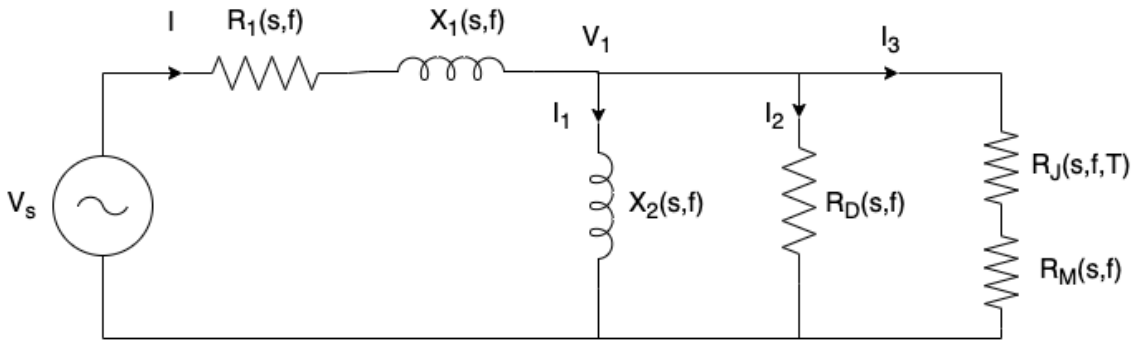


Figure 5.3.4: Equivalent circuit of one phase of an annular linear induction pump used in the detailed circuit model.

The equivalent circuit used in the detailed model is shown in Figure 5.3.4 where

R_1 = primary electrical resistance

R_D = electrical resistance of ducts

R_J = Joule heating resistance

R_M = fluid mechanical work resistance

$X_1 = 2\pi f L_1$ = coil leakage reactance

$X_2 = 2\pi f L_2$ = magnetizing reactance

L_1 = coil leakage inductance

L_2 = magnetizing inductance

f = power source frequency

Note that all circuit parameters in the detailed circuit model can be functions of both s and f. These functionalities are specified using *FUNCTION blocks*.

Using this circuit, Ψ in Eq. (5.3-165) is

$$\Psi = \frac{R_m R_D^2 X_2^2}{(R_j + R_m)^2 R_D^2 + (R_J + R_m + R_D)^2 X_2^2} \tag{5.3-172}$$

Steady-state initialization of the detailed circuit model starts by evaluating the initial frequency specified through *fFuncID*. Using this, the initial slip is calculated with the user-provided synchronous speed. Having both frequency and slip, the respective FUNCTION blocks for each of the circuit parameters are evaluated at t = 0.

If temperature adjustment of circuit parameters is requested, R_J at each timestep is adjusted to R'_J using the proportionality relation

$$R'_J = C\rho_r(T) \tag{5.3-173}$$

where C is a proportionality constant determined using the value of R_J at t = 0 along with *TR*.

5.3.4.3.3.2 Simple Circuit Model

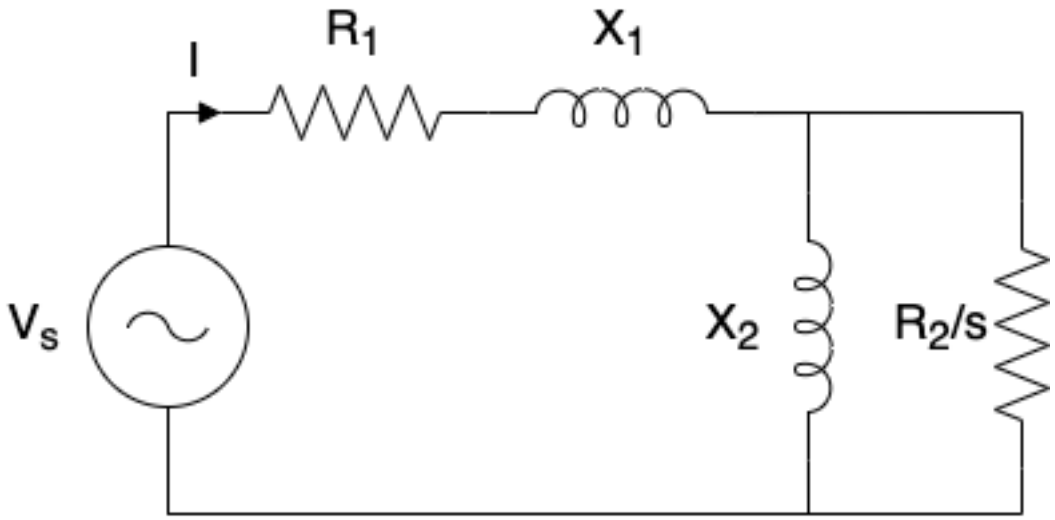


Figure 5.3.5: Equivalent circuit of one phase of an annular linear induction pump used in the simple circuit model.

The equivalent circuit used in the simple model is shown in Figure 5.3.5 where R_2 is a resistance representing both Joule and mechanical power. Note that all circuit parameters in the simple circuit model are assumed constant.

Using this circuit, Ψ in Eq. (5.3-165) is

$$\Psi = \frac{R_2(1-s)}{s\left(\frac{R_2^2}{X_2^2 s^2} + 1\right)} \tag{5.3-174}$$

Determination of circuit parameters during steady-state initialization of the simple circuit model consists of an inversion process using "rated" conditions specified by the user. In this way, the user is not required to know specific electrical/geometric details of the pump, but rather specifies only operating characteristics instead, making the simple model easier to use when pump details are unknown. In this case, "rated" conditions can be any operating state of the pump, though all specified rated conditions for a given pump should be for a fixed operation point (i.e. it is not permissible to specify rated flow at one voltage and rated current at another).

The equations used for determining the circuit parameters are

$$R_2 = \frac{\Delta P_{P,R}(A_R v_s \frac{s_R^2-1}{s_R})}{3I_R^2 - 3I_S^2 \frac{\Delta P_{P,R}}{\Delta P_S s_R}} \tag{5.3-175}$$

$$X_{2,R} = \frac{R_2}{\sqrt{\frac{3I_S^2 R_2}{A_R v_{s,R} \Delta P_{P,S}} - 1}} \tag{5.3-176}$$

$$R_1 = \frac{\frac{\Delta P_{P,R} Q_R}{3I_R^2 \eta_R}(R_2^2 + X_2^2 s_R^2) - s_R R_2 X_2^2}{R_2^2 + X_2^2 s_R^2} \tag{5.3-177}$$

$$X_{1,R} = \left[\left(\frac{V_R}{I_R}\right)^2 - \left(R_1 + \frac{s_R R_2 X_2^2}{R_2^2 + s_R^2 X_2^2}\right)^2\right]^{0.5} - \frac{R_2^2 X_2}{R_2^2 + X_2^2 s_R^2} \tag{5.3-178}$$

where the subscripts R and S indicate rated and stall (i.e. zero flow) conditions, respectfully.

iDPOPT is used to adjust Δp_R for friction losses if requested by the user. Then all values in Eq. (5.3-175) - Eq. (5.3-178) can be deduced from user input except for I_S. Therefore these equations are solved using a fixed-point iteration on I_S until convergence to within *epsIs* is achieved. Once converged, L_1 and L_2 are deduced using f_R. Steady-state circuit reactances are then calculated using frequency at t = 0 specified through *f0*. Note that the inversion process used to arrive at Eq. (5.3-175) - Eq. (5.3-178) makes the determinination of circuit parameters in the simple circuit model highly sensitive to the rated conditions specified. For this reason, a set of suggested values to represent a realistic pump are provided Table 5.3.10.

If temperature adjustment of circuit parameters is requested, R_2 at each timestep is adjusted to R_2' using the proportionality relation

$$R_2' = C\rho_r(T) \tag{5.3-179}$$

where C is determined as outlined for the *detailed circuit model*.

5.3.4.3.3.3 Model Inputs

Inputs for equivalent circuit models consist of data entered in both *EMPUMP blocks* and in *standard input records*. In addition to those inputs required for all liquid elements in *INPMR4* and *PMR4IN*, standard input records related specifically to equivalent circuit pumps are highlighted in Table 5.3.7.

Table 5.3.7: INPMR4 inputs for Equivalent Circuit models.

Variable	Description
ITYPEL(IELL)	Liquid element type of element IELL, specify 5 for a non-annular pump or 15 for an annular .
IELPMP(IPMP)	Element number corresponding to pump IPMP.
IEMPMP(IPMP)	Option to specify that pump IPMP uses the equivalent circuit EM pump model (IEMPMP = -3).
ILRPMP(IPMP)	*EMPUMP block* number containing information for equivalent circuit pump IPMP.

For pumps specified as annular (*ITYPEL* = 15), a number of additional inputs are required in *INPMR4* and *PMR4IN*.

The required EMPUMP block inputs are highlighted below by those which are *common to all models*, those which are *specific to the detailed circuit model*, and those which are *specific to the simple circuit model*. For the simple circuit model, suggested values are provided to represent the pump from Ref. *[5-11]*.

Table 5.3.8: EMPUMP block inputs common to all Equivalent Circuit model options.

Keyword	Description	Units	Type	Range
VFuncID	*FUNCTION block* number corresponding to $V_s(t)$, the normalized pump voltage with time. If omitted, voltage is held constant at steady-state value. If $V_s(0) \neq 1$, table will be normalized to the steady-state value.	–	Integer, optional	0 < VFuncID
fFuncID	*FUNCTION block* number corresponding to $f(t)$, the normalized pump frequency with time.	–	Integer	0 < fFuncID
f0	Initial frequency of the pump.	Hz	Float	0 < f0
fH	Fraction of *pump heating* to be included in thermal calculations. If omitted, pump heating is turned off.	–	Float, optional	0 ≤ fH
fOD	Fraction of *pump heating* to be deposited in outer wall of an annular pump element. Ignored for pipe-type pumps. If omitted, $F_{OD} = \frac{MC_{OD}}{MC_{OD}+MC_{ID}}$, where MC_{OD} is **WALLMC** and MC_{ID} is **WALMC2**.	–	Float, optional	0 ≤ fOD ≤ 1
TR	Temperature of fluid flow through the pump at rated conditions. If omitted, temperature dependence of circuit parameters is turned off.	K	Float, optional	0 ≤ TR
C1	Constant coefficient in Eq. (5.3-171). If any of C1, C2, or C3 are provided, all three defaults are overwritten.	Ω-m	Float, optional	–
C2	Linear coefficient in Eq. (5.3-171). If any of C1, C2, or C3 are provided, all three defaults are overwritten.	Ω-m/C	Float, optional	–
C3	Quadratic coefficient in Eq. (5.3-171). If any of C1, C2, or C3 are provided, all three defaults are overwritten.	Ω-m/C^2	Float, optional	–

Table 5.3.9: Model inputs for the DETAILED option.

Keyword	Description	Units	Type	Range
R1FuncID	*FUNCTION block* number corresponding to $R_1(s,f)$.	–	Integer	0 < R1FuncID
RDFuncID	*FUNCTION block* number corresponding to $R_D(s,f)$.	–	Integer	0 < RDFuncID
RJFuncID	*FUNCTION block* number corresponding to $R_J(s,f)$.	–	Integer	0 < RJFuncID
RMFuncID	*FUNCTION block* number corresponding to $R_M(s,f)$.	–	Integer	0 < RMFuncID
L1FuncID	*FUNCTION block* number corresponding to $L_1(s,f)$.	–	Integer	0 < L1FuncID
L2FuncID	*FUNCTION block* number corresponding to $L_2(s,f)$.	–	Integer	0 < L2FuncID
vs	Synchronous pump speed corresponding to $f(0)$.	m/s	Float	0 < vs

Table 5.3.10: Model inputs for the SIMPLE option.

Keyword	Description	Units	Type	Range	Suggested Value
dPR	Developed pressure over the pump at rated conditions.	Pa	Float	0 < dPR	2.0E5
dPS	Developed pressure over the pump at stall conditions (i.e. zero flow) at rated voltage and frequency.	Pa	Float	0 < dPS	3.0E5
etaR	Fractional pump efficiency at rated conditions.	–	Float	0 < etaR ≤ 1	0.45
IR	Phase current at rated conditions.	A	Float	0 < IR	1700
VR	Phase voltage at rated conditions.	V	Float	0 < VR	510
fR	Power supply frequency at rated conditions.	1/s	Float	0 < fR	12
WR	Mass flow rate through the pump at rated conditions.	kg/s	Float	0 < WR	1333
sR	Slip at rated conditions.	–	Float	0 < sR < 1	0.2
iDPOPT	If iDPOPT ≠ 0, rated and stall input parameters have been adjusted to account for pressure losses. If iDPOPT = 0 or omitted, rated and stall input parameters are based on net pressure rise.	–	Integer, optional	–	1
espIs	Fractional convergence tolerance of the fixed-point iteration to determine stall current. If omitted, epsIS = 10E-8.	–	Float, optional	0 < epsIs	–

Example input showing a detailed-circuit annular pump along with a simple-circuit non-annular pump:

```
INPMR4     3     1     0
#
# NELEMT
    10 : 2        # only two elements
# ITYPEL
    49 : 5 15     # first is a pump, second is annular pump
# NANEL
  1243 : 1        # one element is annular
# IELANE
  1244 : 2        # element 2 is the annular element
# NPUMP
   405 : 2        # two pumps
# IELPMP
   406 : 1 2      # elements 1 and 2 are pumps
# IEMPMP
   418 : -3 -3    # pumps 1 and 2 are equivalent circuit pumps
# ILRPMP
   430 : 1 2      # eq circ pumps correspond to EMPUMP blocks 1 and 2,␣
↪respectively
#
# other integer inputs, including segments, CVs, temperature groups, etc...
#
    -1
#
PMR4IN    18     1     0
#
# element float inputs, including ZOUTEL, XLENEL, AREAEL, etc...
#
# annular element float inputs, including WALMC2,  WALLH2, etc...
#
    -1
#
EMPUMP 1 SIMPLE
    VFuncID = 1
    fFuncID = 1
    f0 = 12
    fH = 0.5          # pump heat is modeled at half strength
    TR = 0            # do not adjust R2 for temperature
    dPR = 0.2E6
    dPS = 0.3E6
    etaR = 0.45
    IR = 1700.
    VR = 850.
    fR = 12.
    WR = 1333
    sR = 0.2
    epsIs = 1E-10
    iDPOPT = 1

    FUNCTION 1 "VoltageAndFrequencyFunction"
        TYPE = TABLE
        SOURCE = 1
    END

    # voltage and frequency hold constant
    TABLE 1 VoltageTable
```

(continues on next page)

(continued from previous page)

```
        x    y
        0    1
        1    1
    END
END

EMPUMP 2 DETAILED
    fFuncID = 1
    VFuncID = 10
    f0 = 12
    fOD = 0.5
    fH = 0        # pump heat is not modeled
    TR = 573.15  # adjust RJ for temperature, using 573.15 K to determine␣
↪proportionality constant
    vs = 12.
    R1FuncID = 2
    RDFuncID = 3
    RJFuncID = 4
    RMFuncID = 5
    L1FuncID = 6
    L2FuncID = 7

    # voltage is halved after 100 seconds
    FUNCTION 10 "VoltageFunction"
        TYPE = TABLE
        TABLE 1 VoltageTable
            x    y
            0    1
            100  0.5
        END
    END

    # frequency doubles after 100 seconds
    FUNCTION 1 "FrequencyFunction"
        TYPE = TABLE
        TABLE 1 FrequencyTable
            x    y
            0      1
            100    2
        END
    END

    # R1 holds constant
    FUNCTION 2 "R1Function"
        TYPE = TABLE
        TABLE 1 SuperLocalTable
            x    y
            0    0.08
            1    0.08
        END
    END

    # RD holds constant
    FUNCTION 3 "RDFunction"
        TYPE = TABLE
```

(continues on next page)

(continued from previous page)

```
        TABLE 1 RDTable
            x    y
            0    10.
            1    10.
        END
    END

    # RD holds constant
    FUNCTION 4 "RJFunction2"
        TYPE = TABLE
        TABLE 1 RJTable
            x    y
            0    0.44
            1    0.44
        END
    END

    # RM holds constant
    FUNCTION 5 "RMFunction"
        TYPE = TABLE
        TABLE 1 RMTable
            x    y
            0    0.7
            1    0.7
        END
    END

    # L1 holds constant
    FUNCTION 6 "L1Function"
        TYPE = TABLE
        TABLE 1 L1Table
            x    y
            0    0.004
            1    0.004
        END
    END

    # L2 holds constant
    FUNCTION 7 "L2Function"
        TYPE = TABLE
        TABLE 1 L2Table
            x    y
            0    0.01
            1    0.01
        END
    END
END
```

5.3.5 Valves

SAS4A/SASSYS-1 contains a simple valve treatment in which the user can specify the valve pressure loss coefficient as a function of time. The pressure drop, Δp_{v}, through a valve is calculated as

$$\Delta p_{\mathrm{v}}(t) = G_2(t)\,\frac{w\,|w|}{2\rho A^2} \tag{5.3-180}$$

where

w = coolant flow rate through the valve

ρ = coolant density

A = nominal valve flow area

G_2 = valve coefficient

The user supplies a table of G_2 as a function of time. The table is used to obtain an average value of G_2 for each step of the transient. With this treatment, a valve is never entirely shut; although G_2 can be made very large so that the flow through the valve becomes very small.

Some care must be taken in setting the input for a valve. The table of G_2 vs. t goes in the DTMPTB and TMPMTB input arrays, but in addition $G_2\,(t=0)$ goes in the input variable G2PRDR (ELL) for the element IELL corresponding to the valve. During the steady-state initialization, G2PRDR may be adjusted by the code if necessary to achieve a steady-state pressure balance. At the end of the steady-state initialization, if G2PRDR does not match the table entry for $t = 0$, then a constant value is added to all of the entries in the DTMPTB table for the valve so that the table does match G2PRDR at $t = 0$. Up to eight valves can be used. Each valve can use its own separate valve coefficient table. It is also possible to use the same table for two or more valves but this option must be used with care. All valves using the same table should have the same steady-state valve coefficients and the same time dependence. Also, if a valve table is adjusted to achieve a steady-state pressure balance for one valve, the adjustment may not be appropriate for other valves using the same table.

5.3.6 Check Valves

SAS4A/SASSYS-1 contains a simple check valve treatment in which the user can specify the valve orifice coefficient as a function of coolant flow rate. Every time step an orifice coefficient (G_2 in Eq. (5.3-47)) is re-set using a user-supplied table of orifice coefficient vs. normalized flow rate, with a flow rate normalized to its steady-state value. Other than the variable orifice coefficient, a check valve is treated the same as a pipe, as described in Section 5.3.3.

5.4 Liquid Temperature Calculations

In Section 5.2, we described how the liquid pressures and liquid mass flow rates are calculated during a time step, ignoring liquid temperature changes and gas mass flow rate changes during the same time step. In this section, we shall describe how liquid temperatures are calculated during a time step while still ignoring gas mass flow rate changes during the time step.

PRIMAR-4 contains liquid temperature calculations for a number of types of components. These include pipes, IHXs, steam generators, bypass channels, a general one-node perfect mixing model for compressible volumes, and a thermal stratification model for the outlet plenum. For the IHX two models are available: a detailed model of both the tube and the shell sides, and a simple table look-up model for the primary side only. Similarly there are two steam generator models available: a detailed model for both the sodium and the water sides, and a simple

table look-up model for the sodium side only. Currently, valves, check valves, and pump impellers are treated as pipes when liquid temperatures are calculated.

For temperature calculations, the elements in a liquid segment are combined into **temperature groups**. Each temperature group contains one or more consecutive elements. All of the elements in a temperature group are treated with the same type of liquid temperature calculation, such as *Pipe Temperatures*, *IHX Temperatures*, and *Bypass Channels*. The number of nodes in a temperature group is defined by the input parameter **NTNODE**, and these are distributed over the elements in that group starting with element **IFSTEL** and ending with element **ILSTEL**.

The Lagrangian calculations used for pipe temperatures are more efficient and accurate if a number of connected pipe elements are strung together into a single temperature group, rather than calculating temperatures for each pipe element separately.

When the detailed IHX model is used, a temperature group is defined only for the primary side; the secondary side inherits the same temperature group. Both the tube-side element and the shell-side element are in the same temperature group, since temperatures on both sides are computed at the same time.

For *table lookup* and *air-dump heat exchanger* elements, the temperature group must contain exactly two nodes to represent the inlet and outlet conditions of the element.

5.4.1 Pipe Temperatures

The pipe temperature model is a slug flow model with heat transfer to the pipe walls, as indicated in Figure 5.4.1. The coolant in a pipe is divided into a number of moving nodes or slugs. The node boundaries move with the coolant flow. All nodes in a pipe have equal volumes except for the first and last nodes. The inlet node size starts at zero and grows as the flow continues until it reaches the size of the other nodes. At that point a new node is started at the inlet. Similarly, the outlet node shrinks and eventually is removed when its volume reaches zero. The temperature in a coolant node changes due to internal heating (see *Direct Element Heating*) or heat transfer with the pipe wall, while the temperature in a wall node may change due to internal heating, heat transfer with the coolant, or heat transfer from the outside of the pipe wall. There is one wall node for each coolant node. One radial node is used in the pipe wall. Heat transfer from the outside of the pipe wall is described in Section 5.4.6 on component-to-component heat transfer. Wall nodes do not move, so the wall node in contact with a given coolant node changes periodically as the coolant node boundaries pass wall nodes.

All of the elements in a pipe temperature group are handled at the same time as if they made a single long pipe. The use of equal coolant volumes for each node determines the locations of the wall nodes. If the region represented by a wall node spans the boundary between two elements, then weighted averages are used to obtain the coolant flow area, A_c, wall perimeter, P_{er}, wall mass, M_w, wall heat capacity, c_w, and wall heat-transfer coefficient, h_w, for the node. The averaging is done so as to conserve coolant volume and wall mass times heat capacity.

The primary loop time step is divided into sub-intervals for the pipe temperatures calculations. The coolant slug is ejected from the end, a new slug is formed at the inlet, and the node indexes for intermediate slugs are increased by one. In subsequent sub-intervals, the coolant moves exactly one node per sub-interval until the end of the primary loop time step is approached. Usually, the coolant does not move exactly an integral number of nodes in a primary loop step, so in the last sub-interval the coolant usually moves only a fraction of a node.

For any node except the inlet node, the heat-transfer equation used for the coolant is:

$$\rho_c c_c A_c \frac{\partial \mathrm{T_c}}{\partial \mathrm{t}} = P_{er} h_{wc} \left(T_w - T_c\right) + q'_c \tag{5.4-1}$$

and that for the wall is

$$M_w c_w \frac{\partial \mathrm{T_w}}{\partial \mathrm{t}} = P_{er} h_{wc} \left(T_c - T_w\right) + \left(\mathrm{hA}\right)_{snk} \left(T_{snk} - T_w\right) + q'_w \tag{5.4-2}$$

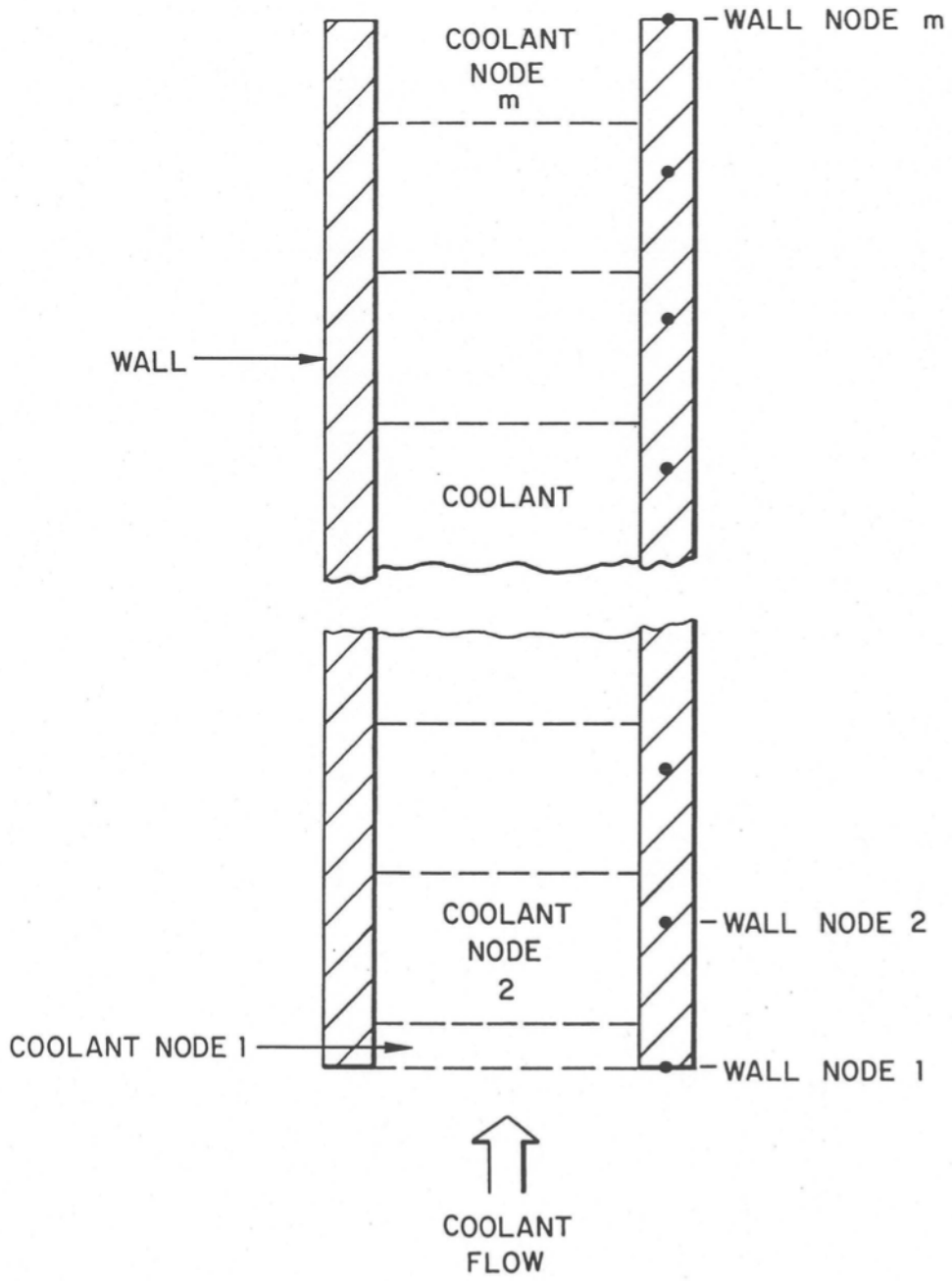


Figure 5.4.1: Pipe Temperature Calculations.

where

T_c = coolant temperature

T_w = wall temperature

ρ_c = coolant density

c_c = coolant specific heat

h_{wc} = effective heat-transfer coefficient between the wall and the coolant

M_w = wall mass per unit length

c_w = specific heat of the wall

P_{er} = wall perimeter, $4A/D_h$

T_{snk} = temperature of a heat sink outside the wall

$(hA)_{snk}$ = heat transfer coefficient times area per unit length for heat transfer to air or liquid sodium outside the pipe wall

q'_c = linear heat generation in the coolant

q'_w = linear heat generation in the wall

The effective heat-transfer coefficient h_{wc} contains a coolant heat-transfer coefficient, h_c, in series with a wall heat-transfer coefficient, h_w:

$$\frac{1}{h_{wc}} = \frac{1}{h_c} + \frac{1}{h_w} \tag{5.4-3}$$

or

$$h_{wc} = \frac{h_c h_w}{h_c + h_w} \tag{5.4-4}$$

The coolant heat-transfer coefficient is calculated as

$$h_c = \frac{k_c}{D_h}\left[C_1\left(\frac{D_h\,|w|\,c_c}{A_c k_c}\right)^{c_2} + C_3\right] \tag{5.4-5}$$

C_1, C_2 and C_3 = user-supplied correlation coefficients

D_h = pipe hydraulic diameter

w = coolant flow rate

k_c = thermal conductivity of the coolant

c_c = specific heat of the coolant

The wall heat-transfer coefficient represents heat transfer from the interior of the wall to the surface in contact with the coolant.

Assuming q'_c and q'_w to be constant over a timestep, finite differencing of Eq. (5.4-1) and Eq. (5.4-2) gives

$$\rho_c c_c A_c \frac{(T_{c6} - T_{c5})}{\delta t} = \frac{P_{er} h_{wc}}{2}\left(T_{w6} - T_{c6} + T_{w5} - T_{c5}\right) + q'_c \tag{5.4-6}$$

and

$$\begin{aligned} M_w c_w \frac{(T_{w6} - T_{w5})}{\delta t} = {} & \frac{P_{er} h_{wc}}{2}\left(T_{c6} - T_{w6} + T_{c5} - T_{w5}\right) \\ & + (hA)_{snk}\left[T_{snk} - \frac{(T_{w6} + T_{w5})}{2}\right] + q'_w \end{aligned} \tag{5.4-7}$$

where

δt = sub-interval time-step size

T_{c5} = coolant temperature at beginning of the sub-interval

T_{c6} = coolant temperature at end of the sub-interval

T_{w5}, T_{w6} = wall temperatures at the beginning and end of the sub-interval

Simultaneous solution of these equations gives

$$T_{c6} = \frac{\left[d_2 - d_1 d_5 \left(M_w c_w + d_3\right)\right] T_{c5} + 2 M_w c_w d_1 d_5 T_{w5} + 2 d_1 d_3 d_5 T_{snk} + \delta t q'_c + d_1 d_5 \delta t q'_w}{d_2 + d_1 d_5 \left(M_w c_w + d_3\right)} \tag{5.4-8}$$

and

$$T_{w6} = d_5 \left(M_w c_w - d_1 - d_3\right) T_{w5} + d_1 d_5 \left(T_{c6} + T_{c5}\right) + 2 d_3 d_5 T_{snk} + d_5 \delta t q'_w \tag{5.4-9}$$

where

$$d_1 = \frac{\delta t}{2} h_{wc} P_{er} \tag{5.4-10}$$

$$d_2 = \rho_c c_c A_c \tag{5.4-11}$$

$$d_3 = \frac{\delta t}{2} \left(hA\right)_{snk} \tag{5.4-12}$$

and

$$d_5 = \frac{1}{d_1 + M_w c_w + d_3} \tag{5.4-13}$$

For the inlet node, the wall temperature calculation is the same as that used for the other nodes, but the coolant temperature calculation is different, since new coolant is being added to the node. The basic equation used for the coolant temperature is

$$L_n \rho_c c_c A_c \frac{\partial}{\partial t} \left(f_r T_c\right) = L_n \rho_c c_c A_c \overline{T}_{in} \frac{\partial f_r}{\partial t} + L_n h_{wc} P_{er} f_r \left(T_w - T_c\right) + f_r q'_c \tag{5.4-14}$$

where

L_n = length of a full node at the inlet

f_r = fraction of a full node at the inlet

$f_r L_n$ = current length of the inlet node

$\overline{T}_{in}$ = pipe inlet temperature

After finite differencing, this equation becomes

$$\begin{aligned} &\rho_c c_c A_c f_{r6} T_{c6} - \rho_c c_c A_c f_{r5} T_{c5} = \rho_c c_c A_c \overline{T}_{in} \left(f_{r6} - f_{r5}\right) \\ &+ \frac{\delta t}{2} h_{wc} P_{er} \left[f_{r5} \left(T_{w5} - T_{c5}\right) + f_{r6} \left(T_{w6} - T_{c6}\right)\right] + f_{r6} q'_c \end{aligned} \tag{5.4-15}$$

where

f_{r5} = f_r at beginning of step

f_{r6} = f_r at end of step.

The simultaneous solution of Eq. (5.4-15) and Eq. (5.4-7) gives

$$T_{c6} = \left\{ \left(d_2 f_{r5} - d_1 f_{r5} + d_1^2 d_5 f_{r6} \right) T_{c5} + d_1 \left[f_{r5} + f_{r6} d_5 \left(M_w c_w - d_1 - d_3 \right) \right] T_{w5} \right. \\ \left. + d_2 \left(f_{r6} - f_{r5} \right) \overline{T}_{in} + 2 f_{r6} d_1 d_3 d_5 T_{snk} + f_{r6} \delta t q'_c + f_{r6} d_1 d_5 \delta t q'_w \right\} / \\ f_{r6} \left[d_2 + d_1 d_5 \left(M_w c_w + d_3 \right) \right] \tag{5.4-16}$$

and Eq. (5.4-9) is again used for the wall temperature.

If flow in the pipe reverses direction, then the temperature calculations are the same, except that the outlet node becomes the inlet node and the inlet node becomes the outlet node.

5.4.1.1 Eulerian Calculations

The slug flow pipe temperature model described above is a LaGrangian treatment that avoids the spurious numerical diffusion that results from typical Eulerian treatments. On the other hand, there are situations in which this treatment requires significantly more computing time than a Eulerian treatment would; and in many of these situations the effects of numerical diffusion would be small, so there is little gain from the time consuming LaGrangian treatment. The Eulerian computation time per subinterval is comparable to the LaGrangian computation time per subinterval, but the LaGrangian time step subinterval size is limited by the restriction that the coolant can not be allowed to move more than one node per subinterval, whereas no such restriction applies in the Eulerian case. Therefore, there is no advantage to a Eulerian treatment if the coolant flow rates are small and the time step sizes are small; but a Eulerian treatment is much faster than a LaGrangian treatment if the coolant flow rate is high and time steps are large enough that the coolant moves many nodes per time step. Therefore, a Eulerian speed-up option has been added to the pipe temperature calculations in the code.

The Eulerian speed-up option can be especially useful in the null transient used for steady-state initialization when component-to-component heat transfer is used. In this case, the temperature time constants are often large, requiring a long null transient to obtain converged temperatures. The temperature solution is numerically stable for large time steps, so one would use a large time step size in the null transient to reduce computing time; but much of the benefit from a large time step size is nullified if a LaGrangian pipe temperature calculation limits the subinterval size to a small value. Also, the numerical diffusion from a Eulerian solution is small or non-existent in a steady-state pipe temperature result, so there is no reason not to use the Eulerian treatment in the null transient.

There are three options for using the Eulerian speed-up. The default option is to always us only the LaGrangian treatment. The second option is to use the Eulerian speed-up in the steady-state null transient but not in the regular transient. The third option is to use the Eulerian speed-up both in the null transient and in the regular transient. In any case, the LaGrangian calculation is used for small time steps in which the coolant will move less than two nodes. If the Eulerian speed-up is being used for a large time step, then first a LaGrangian subinterval is used to move the coolant to the next node boundary. Next, a Eulerian calculation is used to move the coolant the maximum whole number of nodes that will fit within the time step. Finally, a LaGrangian subinterval is used to finish the time step and move the coolant many fraction of a node remaining. The Eulerian part of the calculation is described in Section 5.16.6.

5.4.1.2 Annular Element Temperatures

An annular element is treated the same as a pipe except that an annular element has two walls in contact with the coolant instead of one. The annular element was added to SAS4A/SASSYS-1 in order to model the coolant flow in an RVACS/RACS system in which a relatively thin annulus of sodium flows between the vessel wall and an inner liner. Significant heat transfer occurs between the sodium in the annulus and both the vessel wall and the inner liner.

For the annular element, the heat transfer equation used for the coolant is

$$\rho_{\mathrm{c}} c_{\mathrm{c}} A_{\mathrm{c}} \frac{\partial \mathrm{T_c}}{\partial \mathrm{t}} = P_{\mathrm{era}} h_{\mathrm{wca}} \left(T_{\mathrm{wa}} - T_{\mathrm{c}}\right) + P_{\mathrm{erb}} h_{\mathrm{wcb}} \left(T_{\mathrm{wb}} - T_{\mathrm{c}}\right) + q'_{\mathrm{c}} \tag{5.4-17}$$

where P_{era}, h_{wca}, and T_{wa} refer to wall a, and P_{erb}, h_{wcb}, and T_{wb} refer to wall b. Eq. (5.4-2) and Eq. (5.4-7) are still applicable to each wall. Finite differencing of Eq. (5.4-17) gives

$$\begin{aligned} \rho_{\mathrm{c}} c_{\mathrm{c}} A_{\mathrm{c}} \frac{\left(T_{\mathrm{c6}} - T_{\mathrm{c5}}\right)}{\delta \mathrm{t}} = {} & \frac{P_{\mathrm{era}} h_{\mathrm{wca}}}{2} \left(T_{\mathrm{wa6}} - T_{\mathrm{c6}} + T_{\mathrm{wa5}} - T_{\mathrm{c5}}\right) \\ & + \frac{P_{\mathrm{erb}} h_{\mathrm{wcb}}}{2} \left(T_{\mathrm{wb6}} - T_{\mathrm{c6}} + T_{\mathrm{wb5}} - T_{\mathrm{c5}}\right) + q'_{\mathrm{c}} \end{aligned} \tag{5.4-18}$$

Simultaneous solution of the finite difference equations for T_{c}, T_{wa}, and T_{wb} gives

$$\begin{aligned} T_{\mathrm{c6}} = \{ & \left[d_2 - d_{1\mathrm{a}} d_{5\mathrm{a}} \left(M_{\mathrm{wa}} c_{\mathrm{wa}} + d_{3\mathrm{a}}\right) - d_{1\mathrm{b}} d_{5\mathrm{b}} \left(M_{\mathrm{wb}} c_{\mathrm{wb}} + d_{3\mathrm{b}}\right)\right] T_{\mathrm{c5}} \\ & + 2 d_{1\mathrm{a}} d_{5\mathrm{a}} \left(M_{\mathrm{wa}} c_{\mathrm{wa}} T_{\mathrm{wa5}} + d_{3\mathrm{a}} T_{\mathrm{snka}} + \frac{\delta t}{2} q'_{\mathrm{wa}}\right) \\ & + 2 d_{1\mathrm{b}} d_{5\mathrm{b}} \left(M_{\mathrm{wb}} c_{\mathrm{wb}} T_{\mathrm{wb5}} + d_{3\mathrm{b}} T_{\mathrm{snkb}} + \frac{\delta t}{2} q'_{\mathrm{wb}}\right) + \delta t q'_{\mathrm{c}} \Big\} \\ & \Big/ \left[d_2 + d_{1\mathrm{a}} d_{5\mathrm{a}} \left(M_{\mathrm{wa}} c_{\mathrm{wa}} + d_{3\mathrm{a}}\right) + d_{1\mathrm{b}} d_{5\mathrm{b}} \left(M_{\mathrm{wb}} c_{\mathrm{wb}} + d_{3\mathrm{b}}\right)\right] \end{aligned} \tag{5.4-19}$$

where q'_{wa} and q'_{wb} refer to linear heat sources for walls a and b, respectively. The solutions for T_{wa} and T_{wb} are the same as Eq. (5.4-9). For the inlet node, simultaneous solution of the finite difference equations gives

$$\begin{aligned} T_{\mathrm{c6}} = \{ & \left[d_2 f_{\mathrm{r5}} - d_{1\mathrm{a}} \left(f_{\mathrm{r5}} - f_{\mathrm{r6}} d_{1\mathrm{a}} d_{5\mathrm{a}}\right) - d_{1\mathrm{b}} \left(f_{\mathrm{r5}} - f_{\mathrm{r6}} d_{1\mathrm{b}} d_{5\mathrm{b}}\right)\right] T_{\mathrm{c5}} \\ & + d_{1\mathrm{a}} \left[f_{\mathrm{r5}} + f_{\mathrm{r6}} d_{5\mathrm{a}} \left(M_{\mathrm{wa}} c_{\mathrm{wa}} - d_{1\mathrm{a}} - d_{3\mathrm{a}}\right)\right] T_{\mathrm{wa5}} \\ & + d_{1\mathrm{b}} \left[f_{\mathrm{r5}} + f_{\mathrm{r6}} d_{5\mathrm{b}} \left(M_{\mathrm{wb}} c_{\mathrm{wb}} - d_{1\mathrm{b}} - d_{3\mathrm{b}}\right)\right] T_{\mathrm{wb5}} \\ & + d_2 \overline{T}_{\mathrm{in}} \left(f_{\mathrm{r6}} - f_{\mathrm{r5}}\right) + 2 f_{\mathrm{r6}} \left[d_{1\mathrm{a}} d_{3\mathrm{a}} d_{5\mathrm{a}} T_{\mathrm{snka}} + d_{1\mathrm{b}} d_{3\mathrm{b}} d_{5\mathrm{b}} T_{\mathrm{snkb}}\right] \\ & + f_{\mathrm{r6}} \left[\delta t q'_{\mathrm{c}} + d_{1\mathrm{a}} d_{5\mathrm{a}} \delta t q'_{\mathrm{wa}} + d_{1\mathrm{b}} d_{5\mathrm{b}} \delta t q'_{\mathrm{wb}}\right] \} \\ & \Big/ f_{\mathrm{r6}} \left[d_2 + d_{1\mathrm{a}} d_{5\mathrm{a}} \left(M_{\mathrm{wa}} c_{\mathrm{wa}} + d_{3\mathrm{a}}\right) + d_{1\mathrm{b}} d_{5\mathrm{b}} \left(M_{\mathrm{wb}} c_{\mathrm{wb}} + d_{3\mathrm{b}}\right)\right] \end{aligned} \tag{5.4-20}$$

5.4.1.3 Direct Element Heating

During operation, heat may be generated and/or removed in a variety of different components. For example, during pump operation, heat may be generated in both the working fluid and in the pump structure itself. This phenomenon may strongly impact system temperatures under certain conditions, and therefore SAS provides users with flexibility when modeling it.

Direct element heating is currently supported in all pipe-like elements. Pipe-like elements include: pipes, pumps, check valves, valves, annular pipes, and annular pumps. Two mechanisms are in place for defining direct element heat, time-dependent functions and built-in pump heat models.

5.4.1.3.1 Time-Dependent Element Heating

Time-dependent functions can be assigned to individual elements to represent direct coolant and direct structure heating. User-defined functions $f_{c,k}(t)$ and $f_{w,k}(t)$ are associated with specific elements, k, by the tables **CoolHeatTableID** and **WallHeatTableID**, respectively.

$$q'_c = \frac{f_{c,k}(t)}{L_k} \tag{5.4-21}$$

where

$f_{c,k}$ = user-defined, time-dependent function for element k coolant (W).

L_k = length of element k

and

$$q'_w = \frac{f_{w,k}(t)}{L_k} \tag{5.4-22}$$

where

$f_{w,k}$ = user-defined, time-dependent function for element k wall (W).

Descriptions of the CoolantHeat Table and the WallHeat Table are given in Table 5.4.1 and Table 5.4.2, repectively. Sample input is shown in Listing 5.4.1 and Listing 5.4.2.

Table 5.4.1: Input in a Direct Coolant Heat Table

Column Label	Description	Range
`iEll`	Element ID.	$1 \leq$ `iEll` $\leq$ *NELEMT*
`fID`	ID of function defining direct coolant heat for element `iELL`. The output of this function should be in Watts.	

Table 5.4.2: Input in a Direct Wall Heat Table

Column Label	Description	Range
`iEll`	Element ID. If `iEll` is greater than 1000, the second wall of `iEll`-1000 is identifeid.	$1 \leq$ `iEll` $\leq$ *NELEMT*
`fID`	ID of function defining direct wall heat for element `iELL`. The output of this function should be in Watts.	

Listing 5.4.1: Sample Direct Coolant Heat Table Input

```
TABLE <id> <name>
    iEll   fID
    3        1   # Function 1 is assigned to the coolant of Element 3
    4      100   # Function 100 is assigned to the coolant of Element 4
END
```

Listing 5.4.2: Sample Direct Wall Heat Table Input

```
TABLE <id> <name>
    iEll   fID
    1003   200   # Function 200 is assigned to INNER wall of annular element 3.
    3       10   # Function 10 is assigned to the wall of element 3.
    4       11   # Function 11 is assigned to the wall of element 4.
END
```

5.4.1.3.2 Pump Heat

Pump heating is implemented by calculating the contributions to linear heat rates q'_{c}, q'_{w}, q'_{wa}, and q'_{wb} in the equations above. Currently this treatment is limited only to *equivalent circuit pump models*, where the scaling factor *fH* is used to control the fraction of power dissipated in various resistors representing fluid and wall components.

Detailed Equivalent Circuit Model

For reference to the variables cited below, see the *governing circuit*. Power dissipated by R_1 and R_D in each phase is deposited into the element wall while power disspated in R_J is deposited in the fluid. For nodes of element k that corresponds to an equivalent circuit pump, this is expressed as

$$q'_{\text{c}} = \frac{3 f_H I_3^2 R_J}{L_{\text{k}}} \tag{5.4-23}$$

where

L_{k} = length of element k

I_3 = current through resistor R_J

and

$$q'_{\text{w}} = \frac{3 f_H \left(I^2 R_1 + I_2^2 R_D \right)}{L_{\text{k}}} \tag{5.4-24}$$

If the pump element is annular (see *ITYPEL*), users can control how much of the wall heat is put into each of q'_{wa} and q'_{wb} as

$$\begin{aligned} q'_{\text{wa}} &= f_{OD} q'_{\text{w}} \\ q'_{\text{wb}} &= (1 - f_{OD}) q'_{\text{w}} \end{aligned} \tag{5.4-25}$$

where f_{OD} is *fOD*.

Simple Equivalent Circuit Model

For reference to the variables cited below, see the *governing circuit*. Power dissipated by R_1 in each phase is deposited into the element wall as

$$q'_{\text{w}} = \frac{3 f_H I^2 R_1}{L_{\text{k}}} \tag{5.4-26}$$

No power is deposited into the coolant in the simple circuit model.

If the pump element is annular (see *ITYPEL*), users can control how much of the wall heat is put into each of q'_{wa} and q'_{wb} as

$$\begin{aligned} q'_{\text{wa}} &= f_{OD} q'_{\text{w}} \\ q'_{\text{wb}} &= (1 - f_{OD}) q'_{\text{w}} \end{aligned} \tag{5.4-27}$$

where f_{OD} is *fOD*.

5.4.2 Heat Exchangers: Detailed Options

5.4.2.1 Introduction

The Intermediate Heat Exchanger (IHX) and Primary Heat Exchanger (PHX) are characterized by a shell, a primary coolant channel, a tube, and an secondary coolant channel. Under normal conditions the coolant flows down through the primary channel and up through the secondary channel. Flow reversal is included, so that the flow can be either way in either channel. Also, a slant-height parameter is used to permit the flow path through the tube side to be longer than the flow path through the shell side, and a fouling-factor is included to allow for reduced heat transfer in the primary and intermediate flow channels.

The IHX and PHX are modeled as shown in Figure 5.4.2. The shell, primary coolant channel, tube, and secondary coolant channel are divided into between one and 62 vertical sections. The temperatures in the coolants are calculated at the interfaces of the vertical sections, whereas the shell and tube temperatures are calculated at the centers of the vertical sections. In the case of an IHX, the secondary channel is connected at the inlet and outlet to the intermediate loop. In the case of the PHX, the secondary channel is an independent element whose boundary conditions are defined by the user.

The primary flow rate and temperatures at the beginning of the time step as well as the inlet temperatures at the end of the time step are taken from COMMON blocks. Similarly, for an IHX the secondary flow rate and temperatures are taken from COMMON blocks. For a PHX, the secondary flow rate and inlet temperature at the end of the time step are taken from user provided functions. A set of heat transfer equations is set up and solved for the temperatures at the end of the time step. In addition, gravity heads for both the primary and secondary flow channels are calculated using the resulting temperature distributions. The final values are then stored in COMMON blocks.

5.4.2.2 Basic Equations

In the configuration shown in Figure 5.4.2, a heat balance equation is written for every vertical section of the heat exchanger. Thermal conduction is ignored vertically. The outside surface of the shell can be in thermal contact with a heat sink representing other components or air, as discussed in Section 5.4.6. The heat balance for a vertical section of the shell, which is in thermal contact with the adjacent section of the shell-side liquid, is

$$(\rho c)_{\mathrm{SH}} \Delta z P_{\mathrm{s}} d_{\mathrm{SH}} \frac{\partial \mathrm{T}_{\mathrm{SH}}}{\partial \mathrm{t}} = \Delta z P_{\mathrm{s}} H_{\mathrm{s}} \left(\overline{T}_{\mathrm{CS}} - T_{\mathrm{SH}}\right) + \Delta z \left(hA\right)_{\mathrm{snk}} \left(T_{\mathrm{snk}} - T_{\mathrm{SH}}\right) \tag{5.4-28}$$

with

$$\overline{T}_{\mathrm{CS}} = \frac{1}{2}\left[T_{\mathrm{CS}}\left(j\right) + T_{\mathrm{CS}}\left(j+1\right)\right] \tag{5.4-29}$$

$$\frac{1}{H_{\mathrm{s}}} = \frac{1}{h_{\mathrm{CS}}} + \frac{d_{\mathrm{SH}}}{2k_{\mathrm{SH}}} + \frac{1}{h_{\mathrm{FS}}} \tag{5.4-30}$$

$$h_{\mathrm{CS}} = \frac{\overline{k}_{\mathrm{CS}}}{D_{\mathrm{h}}}\left[C_1 \left(\frac{D_{\mathrm{h}} \mid w_{\mathrm{s}} \mid \overline{c}_{\mathrm{CS}}}{A_{\mathrm{CS}} \overline{k}_{\mathrm{CS}}}\right)^{C_2} \left(\frac{\overline{c}_{\mathrm{CS}} \overline{\mu}_{\mathrm{CS}}}{\overline{k}_{\mathrm{CS}}}\right)^{C_4} + C_3\right] \tag{5.4-31}$$

where

$(\rho c)_{\mathrm{SH}}$ = the shell density times specific heat, **RCSHHX**

Δz = the height of the shell section

P_{S} = the perimeter between the shell and coolant, **PERSPX**

d_{SH} = the shell thickness, **DSHIHX**

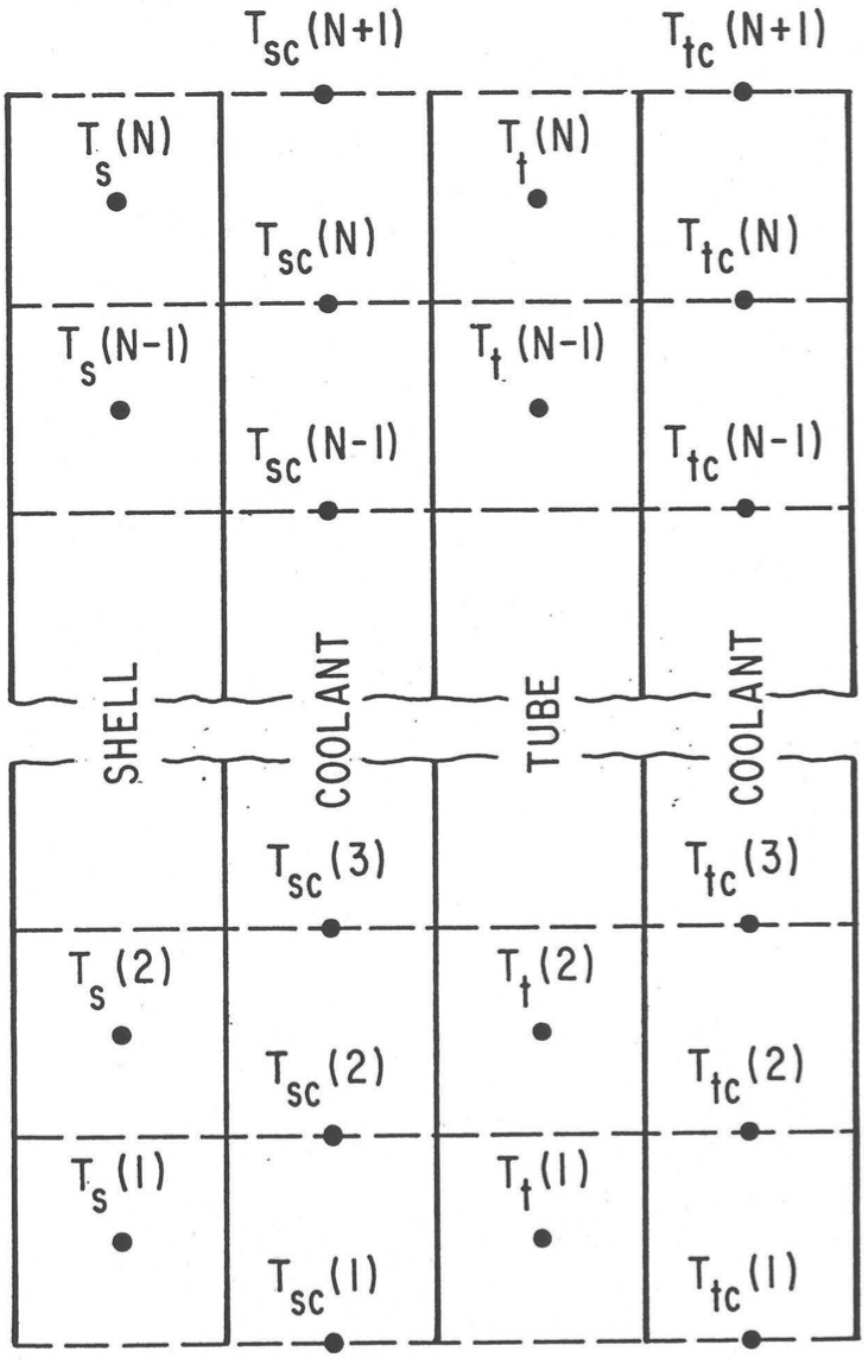


Figure 5.4.2: Intermediate Heat Exchanger Schematic.

T_{SH} = the temperature of the shell section

T_{snk} = the temperature of a heat sink outside the shell

$(hA)_{snk}$ = the heat transfer coefficient times area per unit height for heat transfer to the sink

$T_{CS}(j)$ = the shell-side coolant temperature at the lower end of the vertical section

$T_{CS}(j+1)$ = the shell-side coolant temperature at the upper end of the vertical section

k_{SH} = the shell thermal conductivity, **XKSHHX**

h_{CS} = shell-side coolant heat transfer coefficient

h_{FS} = user-supplied shell-side coolant channel fouling factor, **HFPIHX**

$\overline{k}_{CS}$ = the primary coolant thermal conductivity at the temperature $\overline{T}_{CS}$

$\overline{c}_{CS}$ = the primary coolant specific heat at the temperature $\overline{T}_{CS}$

$\overline{\mu}_{CS}$ = the primary coolant viscosity at the temperature $\overline{T}_{CS}$

$|w_s|$ = the absolute value of the shell-side coolant mass flow rate

D_h = the shell-side coolant channel hydraulic diameter, **DHELEM**

A_{CS} = the shell-side coolant flow area, **AREAEL**

C_1, C_2, C_3 , C_4 = user-supplied correlation coefficients, **C1IHX**, **C2IHX**, **C3IHX**, **C4IHX**

The shell density times specific heat and shell thermal conductivity are user-defined constants independent of temperature, whereas the liquid coolant thermal conductivity viscosity and specific heat are evaluated at the average coolant temperature of a vertical section. The fouling factor is a one-parameter effective film coefficient modeling of the heat transfer in a fouled heat exchanger.

The heat balance for a vertical section of the shell-side coolant, which is in thermal contact with the adjacent sections of the shell and of the tube, is

$$\begin{aligned} A_{CS}\overline{\rho}_{CS}\overline{c}_{CS}\Delta z\frac{\partial \overline{T}_{CS}}{\partial t} + A_{CS}\Delta z\frac{w_S\overline{c}_{CS}}{A_{CS}}\frac{\partial T_{CS}}{\partial z} \\ = \Delta z P_S H_S\left(T_{SH}-\overline{T}_{CS}\right)+\Delta z P_{ST}H_{ST}S\left(T_{TU}-\overline{T}_{CS}\right) \end{aligned} \tag{5.4-32}$$

with

$$\frac{1}{H_{ST}} = \frac{1}{h_{CS}} + \frac{d_{TU}}{2k_{TU}} + \frac{1}{h_{FS}} \tag{5.4-33}$$

where

$\overline{\rho}_{CS}$ = the shell-side coolant density evaluated at the temperature $\overline{T}_{CS}$

w_S = the shell-side coolant mass flow rate

P_{ST} = the perimeter between the shell-side coolant and the tube, **PERPTX**

S = user-supplied slant height ratio of the tube vertical section, **SLANTX**

T_{TU} = the temperature of the tube vertical section

The remaining symbols are the same as defined for the shell.

The slant height ratio enables the user to model the tube-side of the heat exchanger as a coil embedded in the primary coolant. If no slant height is entered in the input, the code sets the slant height to one.

The heat balance for a vertical section of the tube, which is in thermal contact with the adjacent sections of the primary and intermediate coolants is

$$\begin{aligned}(\rho c)_{\mathrm{TU}}\,\Delta z S \frac{1}{2}\left(P_{\mathrm{ST}}+P_{\mathrm{TT}}\right) d_{\mathrm{TU}} \frac{\partial \mathrm{T}_{\mathrm{TU}}}{\partial \mathrm{t}} \\ = \Delta z S P_{\mathrm{ST}} H_{\mathrm{ST}}\left(\overline{T}_{\mathrm{CS}}-T_{\mathrm{TU}}\right)+\Delta z S P_{\mathrm{TT}} H_{\mathrm{TT}}\left(\overline{T}_{\mathrm{CT}}-T_{\mathrm{TU}}\right)\end{aligned} \tag{5.4-34}$$

with

$$\overline{T}_{\mathrm{CT}}=\frac{1}{2}\left[T_{\mathrm{CT}}(j)+T_{\mathrm{CT}}(j+1)\right] \tag{5.4-35}$$

$$\frac{1}{H_{\mathrm{TT}}}=\frac{1}{h_{\mathrm{CT}}}+\frac{d_{\mathrm{TU}}}{2k_{\mathrm{TU}}}+\frac{1}{h_{\mathrm{FT}}} \tag{5.4-36}$$

$$h_{\mathrm{CT}}=\frac{\overline{k}_{\mathrm{CT}}}{D_{\mathrm{h}}}\left[C_1\left(\frac{D_{\mathrm{h}}\,|\,w_{\mathrm{T}}|\;\overline{c}_{\mathrm{CT}}}{A_{\mathrm{CT}}\overline{k}_{\mathrm{CT}}}\right)^{C_2}\left(\frac{\overline{c}_{\mathrm{CT}}\overline{\mu}_{\mathrm{CT}}}{\overline{k}_{\mathrm{CT}}}\right)^{C_4}+C_3\right] \tag{5.4-37}$$

where

$(\rho c)_{\mathrm{TU}}$ = the tube density times specific heat, **RCTUHX**

Δz = the height of the tube section

d_{TU} = the tube thickness, **DTUIHX**

$T_{\mathrm{CT}}(j)$ = the tube-side coolant temperature, lower end of the vertical section

$T_{\mathrm{CT}}(j+1)$ = the tube-side coolant temperature, upper end of the vertical section

k_{TU} = the tube thermal conductivity, **XKTUHX**

h_{CT} = tube-side coolant heat transfer coefficient

h_{FT} = user-supplied tube-side coolant channel fouling factor, **HFIIHX**

$\overline{k}_{\mathrm{CT}}$ = the tube-side coolant thermal conductivity at the temperature $\overline{T}_{\mathrm{CT}}$

$\overline{c}_{\mathrm{CT}}$ = the tube-side coolant specific heat at the temperature $\overline{T}_{\mathrm{CT}}$

$\overline{\mu}_{\mathrm{CT}}$ = the tube-side coolant viscosity at the temperature $\overline{T}_{\mathrm{CT}}$

$|w_{\mathrm{T}}|$ = the absolute value of the tube-side coolant mass flow rate

D_{h} = the tube-side coolant channel hydraulic diameter, **DHELEM**

A_{CT} = the tube-side coolant flow area, **AREAEL**

C_1, C_2, C_3, C_4 = user-supplied correlation coefficients, **C1IHXT**, **C2IHXT**, **C3IHXT**, **C4IHXT**

The tube density times specific heat and tube thermal conductivity are user-defined constants independent of temperature, whereas the tube-side coolant thermal conductivity, viscosity and specific heat are evaluated at the average tube-side coolant temperature of the vertical section. The fouling factor is similar to that for the shell side.

The heat balance for a vertical section of the tube-side coolant, which is in thermal contact with only the adjacent section of the tube, is

$$\begin{aligned}A_{\mathrm{CT}}\overline{\rho}_{\mathrm{CT}}\overline{c}_{\mathrm{CT}}\Delta z S \frac{\partial \overline{\mathrm{T}}_{\mathrm{CT}}}{\partial \mathrm{t}}+A_{\mathrm{CT}}\Delta z S \frac{w_{\mathrm{T}}\overline{c}_{\mathrm{CT}}}{A_{\mathrm{CT}}S}\frac{\partial \mathrm{T}_{\mathrm{CT}}}{\partial \mathrm{z}} \\ = \Delta z S P_{\mathrm{TT}} H_{\mathrm{TT}}\left(T_{\mathrm{TU}}-\overline{T}_{\mathrm{CT}}\right)\end{aligned} \tag{5.4-38}$$

where

$\overline{\rho}_{\mathrm{CT}}$ = tube-side coolant density evaluated at the temperature

w_{T} = the intermediate coolant mass flow rate

The remaining symbols are the same as already defined in this section.

After the primary and intermediate coolant temperatures have been calculated, as described below, the gravity heads for both coolants are calculated by summing terms like the following for each primary and each intermediate coolant section:

$$GH = \overline{\rho} g \Delta z \tag{5.4-39}$$

where

GH = the gravity head for each coolant section

$\overline{\rho}$ = the average coolant density for a coolant section

g = the acceleration of gravity

Δz = the height of the coolant section

5.4.2.3 Finite Difference Equations

If both the shell-side and tube-side coolants did flow through the intermediate heat exchanger in the same direction, the solution would be fairly simple. A set of four simultaneous equations could be solved for each vertical section, and the solutions for all sections could be found by starting at the inlet end and marching the length of the heat exchanger to the exit end. With the flows in opposite directions, however, two approaches are available: iteration or solving a large matrix. Iteration entails guessing one inlet temperature, solving successive sets of 4-by-4 matrices down the length of the IHX, comparing outlet temperatures, and repeating the calculation until outlet temperatures matched. On the other hand, solving a large matrix may involve inverting a 62-by-62 matrix. The iteration method is chosen for the steady-state initialization because that is done only once, and solving the large matrix is chosen for the transient calculation because that is done for each time step.

Eq. (5.4-28), Eq. (5.4-32), Eq. (5.4-34), and Eq. (5.4-38) are converted into difference equations, and the temperature changes during the time step are solved. Time derivatives are replaced by

$$\frac{\partial \mathrm{T}}{\partial \mathrm{t}} = \frac{\Delta T}{\Delta t} \quad ; \quad \Delta T = T_4 - T_3 \tag{5.4-40}$$

where T_3 and T_4 are the temperatures at the beginning and at the end of the time interval Δt. T_1 and T_2 usually denote the beginning and end of a PRIMAR time step, and any subdivision of that time step is denoted by 3 and 4. Space derivatives are taken in the direction of flow. If the flow is down,

$$\frac{\partial \mathrm{T}}{\partial \mathrm{z}} = \frac{T(j) - T(j+1)}{\Delta z} \tag{5.4-41}$$

and if the flow is up,

$$\frac{\partial \mathrm{T}}{\partial \mathrm{z}} = \frac{T(j+1) - T(j)}{\Delta z} \tag{5.4-42}$$

where $T(j)$ and $T(j+1)$ are the temperatures evaluated at the two interfaces of the vertical section of height Δz. The degree of implicitness is introduced by replacing T with

$$T = \theta_1 T_3 + \theta_2 T_4 \tag{5.4-43}$$

where $\theta_1 + \theta_2 = 1$, and is described in Section 5.2.4 and in Section 2.6 in Chapter 2, and T_3 and T_4 are the temperatures at the beginning and end of the time interval Δt.

After making the above substitutions, the equations for the temperature changes during Δt for the j-th vertical section of the IHX for either direction of flow in either coolant channel can be written as:

For the shell,

$$a_1(j)\,\Delta T_{\text{SH}}(j) + a_2(j)\,\Delta T_{\text{CS}}(j) + a_3(j)\,\Delta T_{\text{CS}}(j+1) = a_4(j) \tag{5.4-44}$$

For the shell-side coolant,

$$\begin{aligned} e_1(j)\,\Delta T_{\text{CS}}(j) + e_2(j)\,\Delta T_{\text{SH}}(j-1) + e_3(j)\,\Delta T_{\text{SH}}(j) \\ +e_4(j)\,\Delta T_{\text{TU}}(j-1) + e_5(j)\,\Delta T_{\text{TU}}(j) + e_6(j)\,\Delta T_{\text{CS}}(j-1) \\ +e_7(j)\,\Delta T_{\text{CS}}(j+1) = e_8(j) - e_9(j)\,\Delta T_{\text{CT}}(j-1) - e_{10}(j)\,\Delta T_{\text{CT}}(j) \end{aligned} \tag{5.4-45}$$

For the tube,

$$\begin{aligned} c_1(j)\,\Delta T_{\text{TU}}(j) + c_2(j)\,\Delta T_{\text{CS}}(j) + c_3(j)\,\Delta T_{\text{CS}}(j+1) \\ +c_4(j)\,\Delta T_{\text{CT}}(j) + c_5(j)\,\Delta T_{\text{CT}}(j+1) = c_6(j) \end{aligned} \tag{5.4-46}$$

And for the tube-side coolant,

$$\begin{aligned} f_1(j)\,\Delta T_{\text{CT}}(j) + f_2(j)\,\Delta T_{\text{TU}}(j-1) + f_3(j)\,\Delta T_{\text{TU}}(j) \\ +f_4(j)\,\Delta T_{\text{CT}}(j-1) + f_5(j)\,\Delta T_{\text{CT}}(j+1) \\ = f_6(j) - f_7(j)\,\Delta T_{CS}(j) - f_8(j)\,\Delta T_{\text{CS}}(j+1) \end{aligned} \tag{5.4-47}$$

The expression for each coefficient in terms of the quantities defined in Section 5.4.2.2 are given in Section 5.16.1.

5.4.2.4 Solution

The solution of Eq. (5.4-44) through Eq. (5.4-47) is carried out by a Gaussian elimination scheme using the zeros present in the matrix. The solution algorithm is given in Section 5.16.2. The solution yields the temperature changes throughout the IHX and PHX during the interval Δt, and these values are added to the corresponding temperatures at the beginning of the time interval and the results are stored in COMMON blocks.

5.4.2.5 Steady-State Temperatures

The steady-state temperature distributions in the IHX and PHX are obtained from Eq. (5.4-28), Eq. (5.4-32), Eq. (5.4-34), and Eq. (5.4-38) by setting the time-derivative terms to zero and noting that the temperatures at the beginning and at the end of a time interval are the same. In the steady-state solution, an adiabatic boundary condition is used on the outside of the shell. Therefore, the steady-state shell temperatures are the same as the average shell-side coolant temperatures for each vertical section. As a result, only a 3-by-3 matrix equation must be solved for each vertical section in the iterative solution mentioned at the beginning of Section 5.4.2.3. Also only normal coolant flow, down in the primary and up in the secondary, is considered. Finally, once the equilibrium temperature distributions have been determined, the gravity heads for the primary and intermediate flow channels are computed as indicated in Eq. (5.4-39).

The 3-by-3 matrix that must be solved for the j-th vertical section is:

$$\begin{pmatrix} \alpha_1+\beta_1 & -\beta_1 & 0 \\ -\beta_1 & \alpha_2+\beta_1+\beta_2 & -\beta_2 \\ 0 & -\beta_2 & \alpha_3+\beta_2+\beta_3 \end{pmatrix} \begin{pmatrix} T_{\text{CS}}(j) \\ T_{\text{TU}}(j) \\ T_{\text{CT}}(j) \end{pmatrix} = \begin{pmatrix} D_1 \\ D_2 \\ D_3 \end{pmatrix} \tag{5.4-48}$$

with

$$\alpha_1 = \frac{1}{2}\left[\frac{w_s \bar{c}_{CS}(j)}{\Delta z(j) S} - \beta_1\right]$$

$$\alpha_2 = \beta_1 + \beta_2$$

$$\alpha_3 = -\frac{1}{2}\left[\frac{w_T \bar{c}_{CT}(j)}{\Delta z(j) S} + \beta_2\right]$$

$$\beta_1 = \frac{1}{2} P_{ST} H_{ST}(j)$$

$$\beta_2 = \frac{1}{2} P_{TT} H_{TT}(j)$$

$$\beta_3 = 0$$

$$D_1 = \alpha_1 T_{CS}(j+1)$$

$$D_2 = \beta_1 T_{CS}(j+1) + \beta_2 T_{CT}(j+1)$$

$$D_3 = \alpha_3 T_{CT}(j+1)$$

where

w_S, w_T = the positive steady-state flow rates for the shell and tube sides

The other symbols have the same meanings as in Section 5.4.2.2.

The matrix Eq. (5.4-48) is solved by standard tri-diagonal inversion in subroutine INVRT3.

5.4.2.6 PHX: Boundary Conditions and Coolant Options

The boundary conditions for a PHX are specified using *function blocks*. The inlet temperature for the secondary coolant should be provided in K within the function block **iPHXTID**. In order to ensure a smooth transition from the steady state calculation to the transient calculation, the inlet temperatures defined in **iPHXTID** are corrected. The correction factor is defined as the difference between the user provided steady state temperature and the SAS4A/SASSYS-1 calculated steady state temperature. The flow rate for the secondary coolant should be provided in kg/s within the function block **iPHXWID**. The secondary flowrate can be positive (countercurrent) or negative (cocurrent), but flow reversal can not occur during a transient. A cocurrent configuration requires a null transient calculation and is considered an advanced user option.

The coolant type on the secondary side of a PHX can differ from the coolant type on the primary side of the PHX. The secondary coolant is defined using **iPHXPRP**. The available options for **iPHXPRP** are the same options available for **ICLPRP**. If a value is not provided for **iPHXPRP**, the secondary coolant type is assumed to be the same coolant type as the primary side.

5.4.2.7 HX Input Description

Table 5.4.3: Input in INPMR4 (Block 3) for all HX Models

Variable	Description
NIHX	Number of heat exchangers.
IELIHX	Element number for the primary element in HX IIHX.
ILIHXS	Element number for the secondary element in HX IIHX.
IHXCLC	Model selection parameter for HX IIHX.
ISSIHX	Steady state temperature drop parameter for HX IIHX.

Table 5.4.4: Input in INPMR4 (Block 3) for a PHX Model

Variable	Description
iPHXPRP	Coolant type for secondary element. Optional
iPHXTID	*Function block* ID for secondary element inlet temperature boundary condition.
iPHXWID	*Function block* ID for secondary element flow rate boundary condition.

Table 5.4.5: Input in PMR4IN (Block 18) for all HX Models

Variable	Range	Description
XLENEL	>0.0	Length of the element.
AREAEL	>0.0	Cross-sectional flow area of the liquid element.
DHELEM	>0.0	Hydraulic diameter of liquid element.
ROUGHL	>=0.0	Pipe surface roughness of liquid element
BENDNM	>=0.0	Number of bends in liquid element.
G2PRDR	>=0.0	Initial orifice coefficient of liquid element
WALLMC	>=0.0	Wall heat capacity per unit length of liquid element. Ignored for heat exchanger elements.
WALLH	>=0.0	Wall heat transfer coefficient of liquid element. Ignored for heat exchanger elements.
DTIHX	/-	Optional user specified steady state temperature drop for the primary element within HX.

Table 5.4.6: Input in PMR4IN (Block 18) for the detailed HX Model

Variable	Range	Description
DZSHPX	>=0.0	Fractional axial node heights within HX.
HFPIHX	>=0.0	Fouling heat transfer coefficient for shell side flow within HX.
HFIIHX	>=0.0	Fouling heat transfer coefficient for tube side flow within HX.
SLANTX	>=0.0 <=1.0	Slant-height factor of the tube side flow within HX.
PERSPX	>0.0	Perimeter between shell and shell side coolant within HX.
PERPTX	>0.0	Perimeter between tube and shell side coolant within HX.
PERTIX	>0.0	Perimeter between tube and tube side coolant within HX.
DSHIHX	>0.0	Shell thickness within HX.
DTUIHX	>0.0	Tube thickness within HX.
RCSHHX	>0.0	Volumetric heat capacity of the shell within HX.
RCTUHX	>0.0	Volumetric heat capacity of the tube within HX.
XKSHHX	>0.0	Thermal conductivity of the shell within HX.
XKTUHX	>0.0	Thermal conductivity of the tube within HX.

5.4.3 Bypass Channels

5.4.3.1 Introduction

The purpose of the bypass channel is to model components, like control rods and radial shields, that do not need the detailed treatment of a SAS channel. The bypass channel is then considered a part of the primary loop and is not included in the SAS channel core treatment.

The bypass channel is shown schematically in Figure 5.4.3. It is modeled as two reflectors, A and B, a coolant channel, C, and a duct wall, D. The bypass channel is divided into from 1 to 7 vertical sections, with temperatures of the two reflectors and duct wall taken at the centers of each vertical section and the coolant temperatures taken at the interfaces of each vertical section. The outside surface of reflector A is taken as adiabatic, and heat conduction in the axial direction is neglected. The outside surface of the duct wall can be in contact with a heat sink that represents neighboring subassemblies, as discussed in Section 5.4.1.

Heat sources are included in each of the two regions of reflectors A and B and also in the duct wall, taken as region 3. A vertical power shape for the heat source can be assigned to the sections in the three regions. Two heat sources are included, a neutron heat source and a decay heat source. The neutron heat source arises from fissions caused by neutrons, and is a fraction of the reactor power. The decay heat source arises from fission product decay which produces no neutrons.

A heat balance equation is written for each vertical section of the bypass channel, and the temperature changes during a time step are computed. The choice of the degree of implicitness used in the solution is discussed in Section 5.2.4 and in Section 2.6 in Chapter 2. The temperatures at the beginning of a time step are known, and the coolant mass flow rates at the beginning and at the end of a time step are known from the hydraulics calculations described in Section 5.2. Also, the heat sources during the time step are known. As a result, a set of four simultaneous equations is solved for each vertical section in turn for the length of the bypass channel. In the case of flow reversal, the series of solutions is carried out along the bypass channel in the opposite direction. The temperature changes during the time step are added to the corresponding temperatures at the beginning of the time step, and the final values are then stored in COMMON blocks.

5.4.3.2 Basic Equations

A vertical section of reflector A, with the outside surface assumed adiabatic and with axial thermal conduction ignored, is in thermal contact with only the adjacent section of reflector B. The heat balance equation for the section in A is:

$$(\rho c)_{\mathrm{A}} \Delta z P_{\mathrm{A}} d_{\mathrm{A}} \frac{\partial \mathrm{T_A}}{\partial \mathrm{t}} = \Delta z P_{\mathrm{A}} H_{\mathrm{AB}} \left(T_{\mathrm{B}} - T_{\mathrm{A}}\right) + \Delta z P_{\mathrm{A}} d_{\mathrm{A}} Q_{\mathrm{A}} \tag{5.4-49}$$

with

$$\frac{1}{H_{\mathrm{AB}}} = \frac{d_{\mathrm{A}}}{2k_{\mathrm{A}}} + \frac{d_{\mathrm{B}}}{2k_{\mathrm{B}}} \tag{5.4-50}$$

$$Q_{\mathrm{A}} = \frac{P_{\mathrm{TOT}} Q_{\mathrm{MULT}}}{\Delta z} f_1 \frac{P_{\mathrm{s}}}{P_1} \left(\gamma_{\mathrm{N1}} + \gamma_{\mathrm{D1}}\right), \quad \text{Region 1} \tag{5.4-51}$$

$$Q_{\mathrm{A}} = \frac{P_{\mathrm{TOT}} Q_{\mathrm{MULT}}}{\Delta z} f_2 \frac{P_{\mathrm{s}}}{P_2} \left(\gamma_{\mathrm{N2}} + \gamma_{\mathrm{D2}}\right), \quad \text{Region 2} \tag{5.4-52}$$

where

$(\rho c)_{\mathrm{A}}$ = the reflector A density times specific heat

Δz = the height of the vertical section

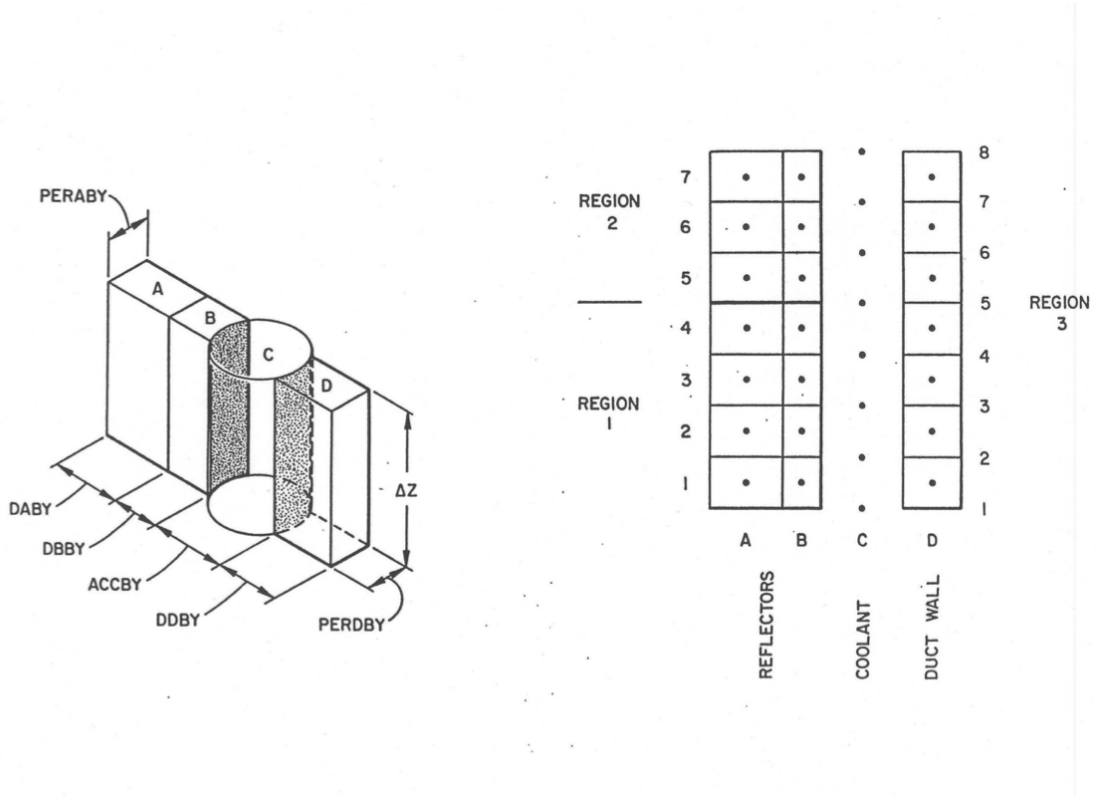


Figure 5.4.3: Bypass Channel Schematic.

P_{A} = the perimeter between reflectors A and B and between B and C

d_{A} = the thickness of reflector A

T_{A} = the temperature of the vertical section of reflector A

T_{B} = the temperature of the vertical section of reflector B

d_{B} = the thickness of reflector B

k_{A} = the thermal conductivity of reflector A

k_{B} = the thermal conductivity of reflector B

Q_{A} = the heat source in the vertical section of reflector A

$P_{\mathrm{TOT}}Q_{\mathrm{MULT}}$ = the reactor power for this time step

f_1 = the fraction of the reactor power distributed in reflector A in Region 1 of the bypass channel

f_2 = the fraction of the reactor power distributed in reflector A in Region 2 of the bypass channel

P_{s} = the power shape assigned to the vertical sections

P_1 = the power shape normalization for Region 1

γ_{N1} = the fraction of the reactor power in Region 1 attributed to neutron heating

γ_{D1} = the fraction of the reactor power in Region 1 attributed to decay heating

P_2 = the power shape normalization for Region 2

γ_{N2} = the fraction of the reactor power in Region 2 attributed to neutron heating

γ_{D2} = the fraction of the reactor power in Region 2 attributed to decay heating

The density times specific heat for reflector A and the thermal conductivities for reflectors A and B are assumed to be independent of temperature. The reactor power for the current time step, $P_{\mathrm{TOT}}Q_{\mathrm{MULT}}$, is described in Chapter 4. The power shape, P_{s}, for each vertical section is an input value, and the code calculates a normalization factor for each region in the bypass channel. The decay heat produced in a section of the reflector or duct wall is taken, in the current version of the code, to be a fraction of the reactor power, just as is the neutron heating. These fractions, the γ's, are input numbers, as are P_{s}, f_1, and f_2.

The heat balance for a vertical section of reflector B, which is in thermal contact with the adjacent sections of reflector A and of the coolant, is

$$\begin{aligned}(\rho c)_{\mathrm{B}}\,\Delta z P_{\mathrm{A}} d_{\mathrm{B}} \frac{\partial \mathrm{T_B}}{\partial \mathrm{t}} = {} & \Delta z P_{\mathrm{A}} H_{\mathrm{AB}}\left(T_{\mathrm{A}} - T_{\mathrm{B}}\right) \\ & + \Delta z P_{\mathrm{A}} H_{\mathrm{BC}}\left(\overline{T}_{\mathrm{C}} - T_{\mathrm{B}}\right) \\ & + \Delta z P_{\mathrm{A}} d_{\mathrm{B}} Q_{\mathrm{B}}\end{aligned} \tag{5.4-53}$$

with

$$\frac{1}{H_{\mathrm{BC}}} = \frac{1}{h_{\mathrm{C}}} + \frac{d_{\mathrm{B}}}{2k_{\mathrm{B}}} \tag{5.4-54}$$

$$h_{\mathrm{C}} = \frac{\overline{k}_{\mathrm{c}}}{D_{\mathrm{h}}}\left[C_1\left(\frac{D_{\mathrm{h}}\left|w_{\mathrm{c}}\right| c_{\mathrm{C}}}{A_{\mathrm{C}}\overline{k}_{\mathrm{c}}}\right)^{C_2} + C_3\right] \tag{5.4-55}$$

$$\overline{T}_{\mathrm{c}} = \frac{1}{2}\left[T_{\mathrm{c}}\left(j\right) + T_{\mathrm{c}}\left(j+1\right)\right] \tag{5.4-56}$$

$$Q_{\mathrm{B}} = \frac{P_{\mathrm{TOT}}Q_{\mathrm{MULT}}}{\Delta z}\left(1 - f_1\right)\frac{P_{\mathrm{s}}}{P_1}\left(\gamma_{\mathrm{N1}} + \gamma_{\mathrm{D1}}\right), \quad \text{Region 1} \tag{5.4-57}$$

$$Q_{\mathrm{B}} = \frac{P_{\mathrm{TOT}} Q_{\mathrm{MULT}}}{\Delta z} \left(1 - f_2\right) \frac{P_{\mathrm{s}}}{P_2} \left(\gamma_{\mathrm{N2}} + \gamma_{\mathrm{D2}}\right), \quad \text{Region 2} \tag{5.4-58}$$

where

$(\rho c)_{\mathrm{B}}$ = the reflector B density times specific heat

Q_{B} = the heat source in the vertical section of reflector B

$T_{\mathrm{c}}\left(j\right)$ = the coolant temperature at the lower end of the vertical section

$T_{\mathrm{c}}\left(j+1\right)$ = the coolant temperature at the upper end of the vertical section

$\overline{k}_{\mathrm{c}}$ = the coolant thermal conductivity at the temperature $\overline{T}_{\mathrm{c}}$

$\overline{c}_{\mathrm{c}}$ = the coolant specific heat at the temperature $\overline{T}_{\mathrm{c}}$

$|w|$ = the absolute value of the coolant mass flow rate

D_{k} = the coolant channel hydraulic diameter

A_{C} = the coolant flow area

C_1, C_2, C_3 = user-supplied correlation coefficients

The remaining symbols are the same as defined above.

The heat balance for a vertical section of the coolant, which is in thermal contact with the adjacent sections of reflector B and the duct wall D, is

$$\begin{aligned} A_{\mathrm{C}} \Delta z \overline{\rho}_{\mathrm{C}} \overline{c}_{\mathrm{C}} \frac{\partial \overline{\mathrm{T}}_{\mathrm{C}}}{\partial \mathrm{t}} + A_{\mathrm{c}} \Delta z \frac{\partial}{\partial \mathrm{z}} \left(\frac{w \overline{c}_{\mathrm{C}}}{A_{\mathrm{C}}} \overline{T}_{\mathrm{C}} \right) &= \Delta z P_{\mathrm{A}} H_{\mathrm{BC}} \left(T_{\mathrm{B}} - \overline{T}_{\mathrm{C}}\right) \\ &\quad + \Delta z P_{\mathrm{D}} H_{\mathrm{CD}} \left(T_{\mathrm{D}} - \overline{T}_{\mathrm{C}}\right) \end{aligned} \tag{5.4-59}$$

with

$$\frac{1}{H_{\mathrm{CD}}} = \frac{1}{h_{\mathrm{C}}} + \frac{d_{\mathrm{D}}}{2k_{\mathrm{D}}} \tag{5.4-60}$$

where

$\overline{\rho}_{\mathrm{C}}$ = the coolant density evaluated at temperature $\overline{T}_{\mathrm{C}}$

w = the coolant mass flow rate

P_{D} = the perimeter between the coolant and the duct wall

d_{D} = the thickness of the duct wall

k_{D} = the thermal conductivity of the duct wall

T_{D} = the temperature of the vertical section of the duct wall

The thermal conductivity of the duct wall is assumed to be independent of temperature. The remaining symbols are the same as already defined.

The heat balance for a vertical section of the duct wall, which, is in thermal contact with the coolant and an external heat sink is

$$\begin{aligned} \left(\rho c\right)_{\mathrm{D}} \Delta z P_{\mathrm{D}} d_{\mathrm{D}} \frac{\partial \mathrm{T}_{\mathrm{D}}}{\partial \mathrm{t}} &= \Delta z P_{\mathrm{D}} H_{\mathrm{CD}} \left(\overline{T}_{\mathrm{C}} - T_{\mathrm{D}}\right) \\ &\quad + \Delta z P_{\mathrm{D}} d_{\mathrm{D}} Q_{\mathrm{D}} + \Delta z \left(hA\right)_{\mathrm{snk}} \left(T_{\mathrm{snk}} - T_{\mathrm{D}}\right) \end{aligned} \tag{5.4-61}$$

with

$$Q_D = \frac{P_{\mathrm{TOT}} Q_{\mathrm{MULT}}}{\Delta z} \frac{P_{\mathrm{s}}}{P_3} \left(\gamma_{\mathrm{N3}} + \gamma_{\mathrm{D3}}\right) \tag{5.4-62}$$

where

$(\rho c)_{\mathrm{D}}$ = the duct wall density times specific heat

Q_{D} = the heat source in the vertical section of the duct wall

P_3 = the power shape normalization for the duct wall

γ_{N3} = the fraction of the reactor power in the duct wall attributed to neutron heating

γ_{D3} = the fraction of the reactor power in the duct wall attributed to decay heating

T_{snk} = the temperature of the heat sink outside the duct wall

$(hA)_{\mathrm{snk}}$ = the heat transfer coefficient times area per unit height for heat transfer to the sink

The remaining symbols are the same as already defined.

5.4.3.3 Finite Difference Equations and Solution

The four equations 5.4-37, 5.4-41, 5.4-47, and 5.4-49 are a set of simultaneous equations that can be solved for one vertical section of the bypass channel. If the coolant flow is upward (normal flow), then the coolant temperature at the bottom of the lowest vertical section is known, and the coolant temperature at the top of the vertical section as well as the reflector and duct wall temperatures are solved for. This process is repeated up the bypass channel for each vertical section. If, on the other hand, the coolant flow is downward (reversed flow), the coolant temperature at the top of the uppermost vertical section is known, and the coolant temperature at the bottom of the section as well as the reflector and duct wall temperatures are solved for. This process is then repeated down the bypass channel for each vertical section.

First, however, the above four equations are converted to finite difference equations and the temperatures at the end of the time step are solved for. The differencing is accomplished by replacing time derivatives by

$$\frac{\partial \mathrm{T}}{\partial \mathrm{t}} = \frac{\Delta T}{\Delta t} \quad ; \quad \Delta T = T_4 = T_3 \tag{5.4-63}$$

where T_3 and T_4 are the temperatures at the beginning and at the end of the time interval Δt. The temperatures T_1 and T_2 usually denote the beginning and end of a PRIMAR time step, and any subdivision of that time step is denoted by 3 and 4. Space derivatives are taken in the direction of flow. If the flow is up,

$$\frac{\partial \mathrm{T}}{\partial \mathrm{z}} = \frac{T\,(j+1) - T\,(j)}{\Delta z} \tag{5.4-64}$$

and if the flow is down,

$$\frac{\partial \mathrm{T}}{\partial \mathrm{z}} = \frac{T\,(j) - T\,(j+1)}{\Delta z} \tag{5.4-65}$$

where $T\,(j)$ and $T\,(j+1)$ are the temperatures evaluated at the two interfaces of the vertical section of height Δz. The degree of implicitness is introduced by replacing T with

$$T = \theta_1 T_3 + \theta_2 T_4 \tag{5.4-66}$$

where $\theta_1 + \theta_2 = 1$, as described in Section 5.2.4 and in Section 2.6 in Chapter 2, and T_3 and T_4 are the temperatures at the beginning and at the end of the time interval Δt.

After making the above substitutions, the equations for the temperatures at the end of the time interval Δt for the j-th vertical section of the bypass channel can be written as:

$$\begin{pmatrix} \alpha_1+\beta_1 & -\beta_1 & 0 & 0 \\ -\beta_1 & \alpha_2+\beta_1+\beta_2 & -\beta_2 & 0 \\ 0 & -\beta_2 & \alpha_3+\beta_2+\beta_3 & -\beta_3 \\ 0 & 0 & -\beta_3 & \alpha_4+\beta_3+\beta_4 \end{pmatrix} \begin{pmatrix} T_{\mathrm{A4}} \\ T_{\mathrm{B4}} \\ T_{\mathrm{C4}} \\ T_{\mathrm{D4}} \end{pmatrix} = \begin{pmatrix} D_1 \\ D_2 \\ D_3 \\ D_4 \end{pmatrix} \tag{5.4-67}$$

with

$$\alpha_1 = (\rho c)_{\mathrm{A}} P_{\mathrm{A}} d_{\mathrm{A}} \tag{5.4-68}$$

$$\alpha_2 = (\rho c)_{\mathrm{B}} P_{\mathrm{B}} d_{\mathrm{B}} \tag{5.4-69}$$

$$\alpha_3 = A_{\mathrm{c}} \overline{\rho}(j)\, \overline{c}(j) + \frac{2\overline{c}(j)}{\Delta z(j)} \theta_2 \Delta t\, |w_4| \tag{5.4-70}$$

$$\alpha_4 = (\rho c)_{\mathrm{D}} P_{\mathrm{D}} d_{\mathrm{D}} \tag{5.4-71}$$

$$\beta_1 = P_{\mathrm{A}} H_{\mathrm{AB}} \Delta t \theta_2 \tag{5.4-72}$$

$$\beta_2 = P_{\mathrm{A}} H_{\mathrm{BC}} \Delta t \theta_2 \tag{5.4-73}$$

$$\beta_3 = P_{\mathrm{A}} H_{\mathrm{DC}} \Delta t \theta_2 \tag{5.4-74}$$

$$\beta_4 = (hA)_{\mathrm{snk}} \Delta t \theta_2 \tag{5.4-75}$$

$$\begin{aligned} D_1 = {} & \left[(\rho c)_{\mathrm{A}} P_{\mathrm{A}} d_{\mathrm{A}} - P_{\mathrm{A}} H_{\mathrm{AB}} \Delta t \theta_1\right] T_{\mathrm{A3}} \\ & + P_{\mathrm{A}} H_{\mathrm{AB}} \Delta t \theta_1 T_{\mathrm{B3}} + \Delta t Q_{\mathrm{A}} \end{aligned} \tag{5.4-76}$$

$$\begin{aligned} D_2 = {} & P_{\mathrm{A}} H_{\mathrm{AB}} \Delta t \theta_1 T_{A3} \\ & + \left[(\rho c)_{\mathrm{B}} P_{\mathrm{B}} d_{\mathrm{B}} - P_{\mathrm{A}} H_{\mathrm{AB}} \Delta t \theta_1 - P_{\mathrm{A}} H_{\mathrm{BC}} \Delta t \theta_1\right] T_{\mathrm{B3}} \\ & + P_{\mathrm{A}} H_{\mathrm{AB}} \Delta t \theta_1 \overline{T}_{\mathrm{C3}} + \Delta t Q_{\mathrm{B}} \end{aligned} \tag{5.4-77}$$

$$\begin{aligned} D_3 = {} & P_{\mathrm{A}} H_{\mathrm{DC}} \Delta t \theta_1 T_{\mathrm{B3}} \\ & + \left[A_{\mathrm{c}} \overline{\rho}(j)\, \overline{c}(j) - \frac{2\overline{c}(j)}{\Delta z(j)} \theta_1 \Delta t\, |w_3| - P_{\mathrm{A}} H_{\mathrm{BC}} \Delta t \theta_1 - P_{\mathrm{D}} H_{\mathrm{DC}} \Delta t \theta_1\right] \overline{T}_{\mathrm{C3}} \\ & + P_{\mathrm{D}} H_{\mathrm{DC}} \Delta t \theta_1 T_{\mathrm{D3}} + \frac{2\overline{c}(j)}{\Delta z(j)} \Delta t \left[\theta_1\, |w_3|\, T_{\mathrm{C3}} + \theta_2\, |w_4 T_{\mathrm{C4}}|\right] \end{aligned} \tag{5.4-78}$$

$$\begin{aligned} D_4 = {} & P_{\mathrm{D}} H_{\mathrm{DC}} \Delta t \theta_1 \overline{T}_{\mathrm{C3}} \\ & + \left[(\rho c)_{\mathrm{D}} P_{\mathrm{D}} d_{\mathrm{D}} - P_{\mathrm{D}} H_{\mathrm{DC}} \Delta t \theta_1 - (hA)_{\mathrm{snk}} \Delta t \theta_1\right] T_{\mathrm{D3}} + \Delta t Q_{\mathrm{D}} \\ & + (hA)_{\mathrm{snk}} \Delta t T_{\mathrm{snk}} \end{aligned} \tag{5.4-79}$$

The matrix Eq. (5.4-67) is solved by standard tri-diagonal inversion in subroutine INVRT3, and the node temperatures of reflectors A and B and duct wall D and the interface temperatures of the coolant are stored in COMMON blocks. In addition, the gravity head of the bypass element is calculated, using the new coolant temperatures, and also stored in COMMON blocks.

5.4.3.4 Steady-State Bypass Channels

The steady-state bypass temperature calculations, performed in subroutine SSBYPS, differ from the transient calculations in that the time derivative terms in Eq. (5.4-49), Eq. (5.4-53), Eq. (5.4-59), and Eq. (5.4-61) are set to zero and in that the steady-state power input into the reflectors and duct wall for each vertical section must be conducted to the coolant in that section. The coolant temperatures are calculated at the interfaces of each vertical section, marching up the bypass channel and iterating over the temperature-dependent thermal properties. From these coolant temperatures, the node temperatures of the reflectors and duct wall are then determined. In addition, the gravity head is calculated from the average coolant density for the whole channel, and the power normalization factors are calculated for all three regions.

5.4.4 Compressible Volumes

The liquid temperature for a compressible volume other than the outlet plenum is computed as a one-point perfect mixing model. If cover gas is present, its contribution to the liquid temperature is ignored because of the small heat capacity of the gas, and the calculation is carried out as though no gas were present. In Eq. (5.3-8) of Section 5.3.1 for the liquid pressure calculation in a compressible volume, the heat flow from the walls was ignored. Here in the liquid temperature calculation it is included in the energy balance equation.

For the temperature calculation, the heat flow is taken as

$$Q\Delta t = \Delta t \left[\theta_{1\mathrm{w}} \left(T_{\mathrm{w}3} - T_3\right) + \theta_{2\mathrm{w}} \left(T_{\mathrm{w}4} - T_4\right)\right] h_{\mathrm{w}} A_{\mathrm{w}} + E_{\mathrm{src}} \tag{5.4-80}$$

where

T_{w} = the compressible volume wall temperature

h_{w} = the effective heat-transfer coefficient

A_{w} = the compressible volume wall area

$\theta_{2\mathrm{w}}$ = the degree of implicitness

T_3 = the liquid temperature at the beginning of the subinterval

T_4 = the liquid temperature at the end of the sub-interval

E_{src} = heat flow to the liquid from other components

Also, the wall temperature is determined by

$$m_{\mathrm{w}} c_{\mathrm{w}} \left(T_{\mathrm{w}4} + T_{\mathrm{w}3}\right) = -Q\Delta t + \Delta t H_{\mathrm{snk}} A_{\mathrm{snk}} \left(T_{\mathrm{snk}} - \theta_{1\mathrm{w}} T_{\mathrm{w}3} - \theta_{2\mathrm{w}} T_{\mathrm{w}4}\right) \tag{5.4-81}$$

where

m_{w} = the mass of the compressible volume wall

c_{w} = the heat capacity of the compressible volume wall

T_{snk} = the temperature of a heat sink representing heat loss to sodium or air outside the compressible volume, as discussed in Section 5.4.6

H_{snk} = the heat transfer coefficient from the compressible volume wall to the heat sink

Combining Eq. (5.4-80) and Eq. (5.4-81) gives

$$\begin{aligned} T_{\mathrm{w}4} = \{T_{\mathrm{w}3} \left[m_{\mathrm{w}} c_{\mathrm{w}} - \theta_{1\mathrm{w}} \Delta t \left(h_{\mathrm{w}} A_{\mathrm{w}} + H_{\mathrm{snk}} A_{\mathrm{snk}}\right)\right] + \Delta t h_{\mathrm{w}} A_{\mathrm{w}} \left(\theta_{1\mathrm{w}} T_3 + \theta_{2\mathrm{w}} T_4\right) \\ + \Delta t H_{\mathrm{snk}} A_{\mathrm{snk}} T_{\mathrm{snk}}\} \Big/ \left[m_{\mathrm{w}} c_{\mathrm{w}} + \theta_{2\mathrm{w}} \Delta t \left(h_{\mathrm{w}} A_{\mathrm{w}} + H_{\mathrm{snk}} A_{\mathrm{snk}}\right)\right] \end{aligned} \tag{5.4-82}$$

for the wall temperature at the end of the time interval.

The $\sum \overline{w}_{\mathrm{out}} T_{\mathrm{out}}$ term in Eq. (5.3-8) involves a compressible volume liquid temperature averaged over the time step. For the hydraulic calculations in Section 5.2, this term is evaluated using the temperature at the beginning of the step as the average; but for the compressible volume temperature calculations this term is evaluated as

$$\sum \overline{w}_{\mathrm{out}} T_{\mathrm{out}} = \sum_{\mathrm{i}} \theta_1 \left(i\right) \left|w_3 \left(i\right)\right| T_3 + \theta_2 \left(i\right) \left|w_4\right| T_4 \tag{5.4-83}$$

where the summation is over all liquid segments, i, in which the flow is out of the compressible volume. Then

$$\Delta t \left\{\sum \overline{w}_{\mathrm{in}} T_{\mathrm{in}} - \sum \overline{w}_{\mathrm{out}} T_{\mathrm{out}}\right\} = S_3 + S_4 T_4 \tag{5.4-84}$$

where

$$S_3 = \Delta t \sum_{\text{k}} \left[\theta_1(k)\,|w_3(k)|\,T_{3\text{ex}}(k) + \theta_2(k)\,|w_4(k)|\,T_{4\text{ex}}(k)\right] - \Delta t \sum_{\text{i}} \theta_1(k)\,|\,w_3(k)|\,T_3 \tag{5.4-85}$$

$$S_4 = -\Delta t \sum_{\text{i}} \theta_2(i)\,|w_4(i)| \tag{5.4-86}$$

$T_{3\text{ex}}$, $T_{4\text{ex}}$ = the temperature of the liquid leaving liquid segment k at the beginning and end of the time semiinterval

In Eq. (5.4-85) and Eq. (5.4-86) the k summations are over all liquid segments in which the flow is out of the compressible volume. Combining Eq. (5.3-8), Eq. (5.4-82), and Eq. (5.4-84) gives

$$T_4 = \left\{ m_3 T_3 + S_3 + \frac{d_1 d_3 d_6 \left(T_{\text{snk}} - \theta_1 T_3\right)}{c_{\text{l}}} + \frac{E_{\text{src}}}{c_{\text{l}}} + \frac{d_1 d_2 d_5}{c_{\text{l}}} \left(T_{\text{w3}} - \theta_{1\text{w}} T_3\right) \right\} / \left[m_4 - S_4 + \frac{d_1 d_3}{c_{\text{l}}} \left(d_2 + \theta_{2\text{w}} d_6\right) \right] \tag{5.4-87}$$

where

$$d_1 = \Delta t h_{\text{w}} A_{\text{w}} \tag{5.4-88}$$

$$d_2 = m_{\text{w}} c_{\text{w}} \tag{5.4-89}$$

$$d_6 = \Delta t H_{\text{snk}} A_{\text{snk}} \tag{5.4-90}$$

$$d_5 = \frac{1}{d_2 + \theta_{2\text{w}} d_1 + \theta_{2\text{w}} d_6} \tag{5.4-91}$$

$$d_3 = \theta_{2\text{w}} d_5 \tag{5.4-92}$$

E_{src} = a heat source due heat transfer other components the volume liquid, as Section 5.4.6.

In the code, first Eq. (5.4-87) is solved for T_4; and then Eq. (5.4-82) is solved for T_{w4}.

5.4.4.1 Thick-Walled Compressible Volumes

In the traditional CV temperature calculation, described above, the compressible volume wall is assumed to have a uniform or time independent temperature profile. This approximation is valid when the wall thickness is thin and the heat capacity of the wall is much less than the heat capacity of the coolant. In the case of a thick compressible volume wall, i.e. the reactor vessel wall, users have the ability to model the temperature evolution within the wall. This option, referred to as the thick-walled compressible volume modeling option, utilizes the same pressure and flow calculations as the traditional CV, hereafter referred to as the thin-walled CV option, but explicitly accounts for the heat transfer through the wall using a nodal approach, shown in Figure 5.4.4.

A thick-walled compressible volume is divided into s nodes. The first and last node are considered surface nodes, which are half the thickness of the inner nodes. At this time, the material properties within the wall are assumed to be temperature independent and the heat transfer area between each node is constant. The boundary conditions at the inner and outer surface of the wall utilize Newton's law of cooling. It is assumed that no heat generation occurs within the CV wall. The temperature solution of the CV coolant is consistent between the thin-walled and thick-walled implementation.

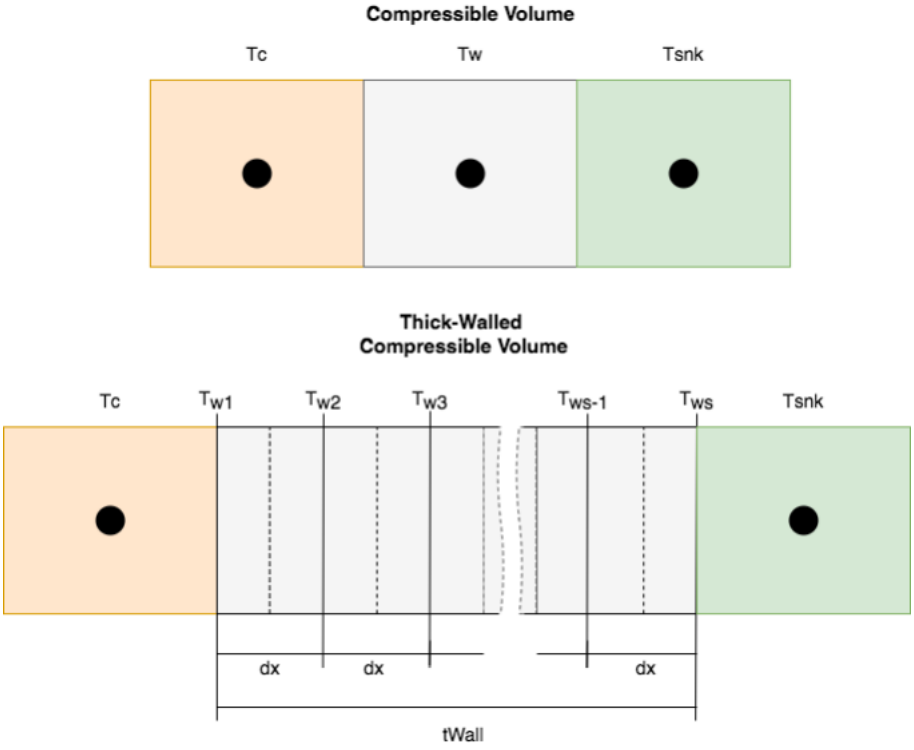


Figure 5.4.4: Geometric representation of a thin-walled (top) and thick-walled (bottom) compressible volume.

5.4.4.1.1 Basic Equations

By combining Eq. (5.3-7) and Eq. (5.4-84), the conservation of energy equation for a thick-walled CV liquid can be written as

$$T_4 m_4 = T_3 m_3 + S_3 + S_4 T_4 + \frac{Q}{c_l}\Delta t \tag{5.4-93}$$

By replacing the average wall temperature, with the wall surface temperature Eq. (5.4-80) can be written as

$$Q\Delta t = \Delta t h_w A_w \left(\theta_1 \left(T_{w1,3} - T_3\right) + \theta_2 \left(T_{w1,4} - T_4\right)\right) \tag{5.4-94}$$

where

$T_{w1,3}$ $T_{w1,4}$ = the compressible volume wall inner surface temperature at the beginning and end of the time step

A similar energy balance can be performed at the inner surface of the compressible volume wall.

$$(mc)_{w1} T_{w1,4} = (mc)_{w1} T_{w1,3} - Q\Delta t + Q_{w1,w2}\Delta t \tag{5.4-95}$$

$$Q_{w1,w2}\Delta t = \Delta t h_{wi} A_w \left(\theta_1 \left(T_{w2,3} - T_{w1,3}\right) + \theta_2 \left(T_{w2,4} - T_{w1,4}\right)\right) \tag{5.4-96}$$

where

$$h_{wi} = \frac{k}{dx} \tag{5.4-97}$$

and

$T_{w2,3}$ $T_{w2,4}$ = the compressible volume wall temperature at node j=2 at the beginning and end of the time step

k = the compressible volume wall thermal conductivity

s = the number of wall nodes

dx = the compressible volume wall node thickness, $\frac{tWall}{s-1}$

mc_{w1} = mc_{ws} = the compressible volume wall surface node heat capacity, $\frac{MCp_{wall}}{2(s-1)}$

The energy balance for the inner wall node, j, will follow a similar pattern

$$(mc)_{wj}T_{wj,4} = (mc)_{wj}T_{wj,3} - Q_{wj-1,wj}\Delta t + Q_{wj,wj+1}\Delta t \tag{5.4-98}$$

$$Q_{wj-1,wj}\Delta t = \Delta t h_{wi}A_w\left(\theta_1\left(T_{wj,3} - T_{wj-1,3}\right) + \theta_2\left(T_{wj,4} - T_{wj-1,4}\right)\right) \tag{5.4-99}$$

where

mc_{wj} = the compressible volume wall inner node heat capacity, $\frac{MCp_{wall}}{s-1}$

Finally, the energy balance for the outer surface node can be written

$$(mc)_{ws}T_{ws,4} = (mc)_{ws}T_{ws,3} - Q_{ws-1,ws}\Delta t + Q_{ws,snk}\Delta t \tag{5.4-100}$$

$$Q_{ws,snk}\Delta t = \Delta t(hA)_{snk}\left(T_{snk} - (\theta_1 T_{ws,3} + \theta_2 T_{ws,4})\right) \tag{5.4-101}$$

Eq. (5.4-93) through Eq. (5.4-101) can be rearranged into a tri-diagonal matrix with s+1 rows. Each transient time step, SAS solves for the temperature of the coolant and wall nodes using standard tri-diagonal inversion in subroutine INVRT3. During steady state initialization, an adiabatic boundary condition is applied at the outer surface of the CV wall. As a result, all wall node temperatures can be set to the coolant temperature.

5.4.4.2 CV Input Description

Table 5.4.7: Input in INPMR4 (Block 3) for all CV Models

Variable	Description
NCVP	Number of CVs in the primary loop.
NCVS	Number of CVs in the secondary loop.
ITYPCV	Compressible volume type.
NCCV	Number of sequential CVs with a common cover gas.
ICCVFS	First CV with common cover gas.

Table 5.4.8: Input in INPMR4 (Block 3) for a thick-walled CV

Variable	Description
ThickWallTableID	*Table ID* for the thick-wall CV table.

Table 5.4.9: Input in PMR4IN (Block 18) for all CV Models

Variable	Range	Description
VOLLGC	>0.0	Total volume, liquid+gas
ALPHAP	/-	Volume pressure expansion coefficient
ALPHAT	/-	Volume temperature expansion coefficient
ZCVL	/-	Reference height for liquid pressure.
AREAIN	>=0.0	Area of liquid-gas interface.
TREFCV	>=0.0	Initial steady-state gas temperature, if present.
BTAPNA	/-	Liquid pressure expansion coefficient
BTATNA	/-	Liquid temperature expansion coefficient
HWALL	/-	Wall-coolant heat transfer coefficient
AWALL	>0.0	CV Wall surface area
CMWALL	>0.0	Heat capacity of the CV wall

Table 5.4.10: Input in the ThickWallTable for the thick-walled CV Model

Column Label	Range	Description
ICV	> 0	ID of CV using a thick-walled approximation.
nNodes	> 2	Number of wall nodes in the thick wall approximation.
kWall	>=0.0	Wall thermal conductivity.
tWall	>=0.0	Wall thickness

Example thick-wall table input:

```
TABLE <id> ThickWallInput
  ICV   nNodes   kWall   tWall
    3       4    32.4    0.05
    6       3    16.2    0.05
END
```

5.4.5 Table Look-Up for Steam Generators or Intermediate Heat Exchangers

Table look-ups for steam generators or intermediate heat exchangers are used when the behavior of these components is known from other sources or when the details of the temperature distributions within the components are not important for the case at hand. The user specifies normalized temperature drops vs. time or outlet temperatures vs. time, as well as elevation of the thermal center vs. time. These boundary conditions can be specified as tables vs. time or as *function blocks*.

The thermal center of the steam generator or intermediate heat exchanger is the point above which the circulating fluid is assumed to have the density corresponding to the inlet temperature and below which it is assumed to have the density corresponding to the outlet temperature.

In the case where the user specifies normalized temperature drops vs. time, the outlet temperature of the steam generator or intermediate heat exchanger is calculated from the inlet temperature by

$$T_{\text{out}}(t) = T_{\text{in}}(t) + f(t)\left[T_{\text{out}}(t=0) - T_{\text{in}}(t=0)\right] \tag{5.4-102}$$

where $f(t)$ is obtained by linear interpolation from the user-supplied table of temperature drops or by evaluation of the user-supplied *function block* which defines temperature drop. Also

$$f(t=0) = 1.0 \tag{5.4-103}$$

In the case where the user specifies outlet temperatures vs. time, the outlet temperature of the steam generator or intermediate heat exchanger is obtained by linear interpolation from the user- supplied table of outlet temperatures or by evaluation of the user-supplied *function block* which defines outlet temperature.

The gravity head is calculated from

$$\Delta p_{gr}(t) = \rho_{in}(t)\, g\, [z_c(t) - z_{in}] + \rho_{out}(t)\, g\, [z_{out} - z_c(t)] \tag{5.4-104}$$

where

$\rho_{in}(t)$, $\rho_{out}(t)$ = the densities of the inlet and outlet fluids at the inlet and outlet temperatures

$z_c(t)$ = the user-supplied elevation of the thermal center as a function of time

z_{in}, z_{out} = the inlet and outlet elevations

g = the acceleration of gravity

5.4.5.1 Table HX Input Description

For model independent HX input refer to Section 5.4.2.7.

Table 5.4.11: Input in INPMR4 (Block 3) for all table look-up IHX/SG models

Variable	Description
IHXCLC	Table number or *function block* ID (ITAB) for HX IIHX
ISGCLC	Table number or *function block* ID (ITAB) for SG ISGN
iHXZID	Table number or *function block* ID (ITAB) defining height of thermal center for HX IIHX
iSGZID	Table number or *function block* ID (ITAB) defining height of thermal center for SG ISGN

If a *function block* ID is provided for an input above, then the corresponding input listed in the tables below is not used.

Table 5.4.12: Input in PMR4IN (Block 18) for all table look-up IHX models

Variable	Range	Description
TMPMTB	>=0.0	Times for **DTMPTB** and **ZCENTR**.
ZCENTR	-	Height of thermal center for an IHX or steam generator.

Table 5.4.13: Input in PMR4IN (Block 18) for a normalized temperature drop table look-up IHX model

Variable	Range	Description
DTMPTB	-	Normalized temperature drop for an IHX or steam generator.

Table 5.4.14: Input in PMR4IN (Block 18) for an outlet temperature table look-up IHX model

Variable	Range	Description
DTMPTB(1,ITAB)	> 1.0	Steady State outlet temperature for the primary element in an IHX or steam generator.
DTMPTB(2:,ITAB)	> 0.0	Outlet temperature for the primary element in an IHX or steam generator.

Table 5.4.15: Input in PMR4IN (Block 18) for a normalized heat rejection table look-up IHX model

Variable	Range	Description
DTMPTB(1,ITAB)	= 1.0	Steady State normalized heat rejection for the primary element in an IHX or steam generator.
DTMPTB(2:,ITAB)	-	Nomalized heat rejection for the primary element in an IHX or steam generator.
DTSIHX	>0.0	The inital temperature difference between the inlet and out of a heat exchanger.
VSIHX1	>0.0	The volume of the inlet region of a heat exchanger.
VSIHX2	>0.0	The volume of the outlet region of a heat exchanger.

5.4.6 Component-to-Component Heat Transfer

SAS4A/SASSYS-1 accounts for heat transfer between components or from a component to a constant temperature heat sink which might represent the air. The code accounts for heat transfer from a core channel duct wall to a core channel duct wall or to a constant temperature heat sink. Also, it accounts for heat transfer from a pipe wall, a bypass channel duct wall, an IHX shell, or a compressible volume wall to the liquid in a compressible volume, to a RVACS (Radiant Vessel Auxiliary Cooling System), or to a constant temperature heat sink. This heat transfer is usually a small effect at normal operating conditions, but it is often important at decay heat power levels. Currently, component-to-component heat transfer is only accounted for in the transient calculations; a null transient, as described in Section 5.9.5, must be run to account for steady-state component-to-component heat transfer.

The component-to-component heat transfer is controlled by user input. The user specifies which components are in contact with which other components, as well as specifying the heat transfer coefficients and areas involved in component to component heat transfer. If the user does not list a component in the component-to-component heat transfer tables, then an adiabatic boundary condition is used on the outside of the component.

The core channel-to-core channel heat transfer is described in Chapter 2. The implementation of component-to-component heat transfer in PRIMAR-4 was carried out in two main steps:

1. Terms T_{snk} and $(hA)_{\text{snk}}$ or H_{snk} A_{snk} were added to the temperature equations for the annular element, the pipe wall Eq. (5.4-2), the bypass channel wall Eq. (5.4-61), the IHX shell Eq. (5.4-28), and the compressible volume wall Eq. (5.4-81) to account for heat flow to adjacent components. Also, a heat source term, E_{src}, was added to Eq. (5.4-80) for the temperature of the liquid in a compressible volume.
2. The T_{snk} terms are re-calculated for each time step of the transient to correspond to the current temperatures of neighboring components. Also, contributions to the E_{src} term are added for each participating component for each time step.

If a pipe wall is in contact with the liquid in a compressible volume I, then T_{snk} for the pipe wall temperature calculation is set equal to the compressible volume liquid temperature at the beginning of each PRIMAR time sub-internal, and $(hA)_{\text{snk}}$ is set to a user-specified value. The contribution to E_{src} from node N of the pipe wall to the compressible volume liquid is calculated as

$$\Delta E_{\text{src}}(I) = [T_{\text{w}}(N) - T_{\text{snk}}](hA)_{\text{snk}} \Delta L(N) \Delta t \tag{5.4-105}$$

where

$T_{\text{w}}(N)$ = the pipe wall temperature at node N

and

$\Delta L(N)$ = the length of node N.

Each node in the liquid element representing the pipe adds a similar contribution to $E_{\text{src}}(I)$. Also, if a bypass channel wall, IHX shell, or the wall of another compressible volume is in contact with the liquid in compressible volume I, it also adds similar contributions to $E_{\text{src}}(I)$.

If a constant temperature heat sink is used for a component, then the user supplies the values for T_{snk} and $(hA)_{\text{snk}}$ or $H_{\text{snk}} A_{\text{snk}}$.

The RVACS model supplies T_{snk} and H_{snk} A_{snk} for components involved in the RVACS, and component temperatures are supplied to the RVACS routine. Additional information about the RVACS component can be found in Section 5.4.7.

5.4.6.1 Element to CV

When in element, **IELHT**, is connected to a CV or a constant heat sink, **IELHT2**, using component to component heat transfer the element wall is connected to the CV liquid or the constant temperature, as shown in Figure 5.4.5.

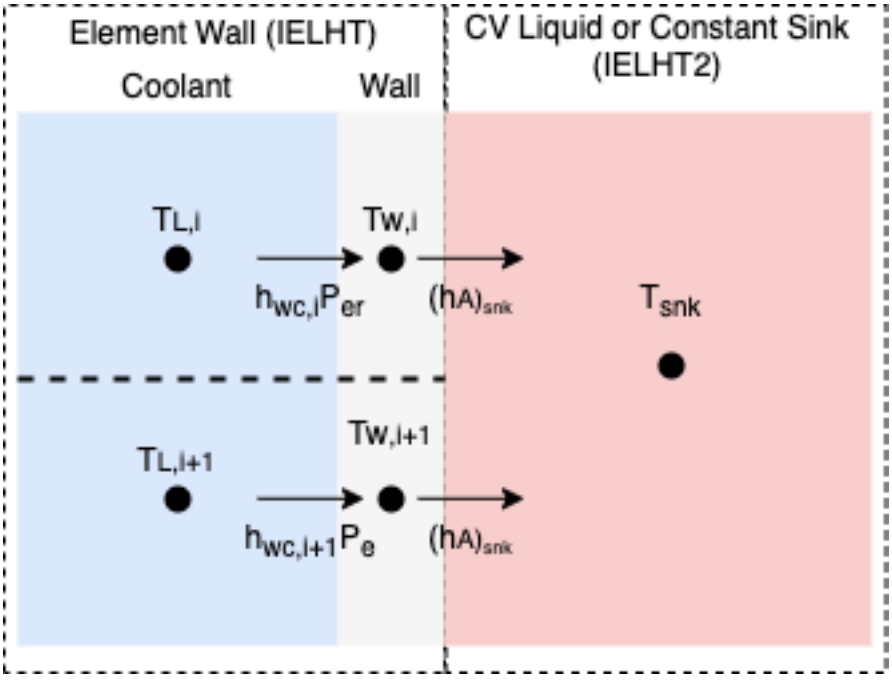


Figure 5.4.5: Component to component heat transfer between an Element and a CV/constant heat sink.

For each element node that is participating in the component to component heat transfer, $(hA)_{\text{snk}}$ is set using **HAELHT**. $h_{wc,i}$ is calculated based on Eq. (5.4-4), using **WALLH** as h_w and P_{er} is 4 **AREAEL** / **DHELEM**.

5.4.6.2 CV to CV

When a CV, **IELHT**, is connected to a CV or a constant heat sink, **IELHT2**, using component to component heat transfer the CV wall is connected to the CV liquid or the constant temperature, as shown in Figure 5.4.6.

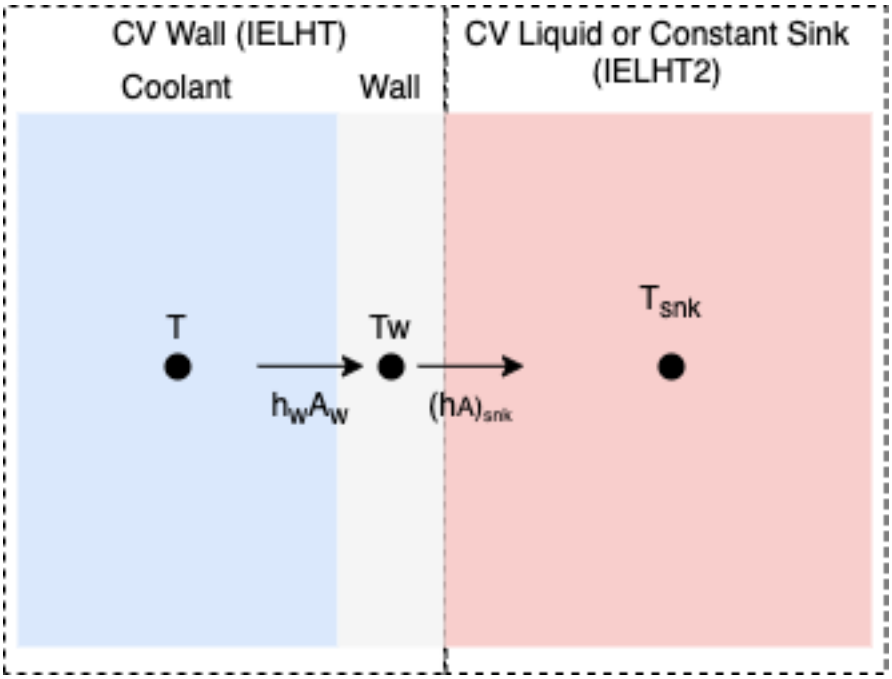


Figure 5.4.6: Component to component heat transfer between a CV and a CV/constant heat sink.

For the CV participating in the component to component heat transfer $(hA)_{\text{snk}}$ is set using **HAELHT** and h_w is **HWALL**, and A_w is **AWALL**.

5.4.6.3 Element to Element

When an element, **IELHT**, is connected to an element, **IELHT2**, using component to component heat transfer the element wall is connected to the element wall, as shown in Figure 5.4.7.

For each element node that is participating in the component to component heat transfer, $(hA)_{\text{snk}}$ is set using **HAELHT**. $h_{wc,i}$ is calculated based on Eq. (5.4-4), using **WALLH** as h_w and P_{er} is 4 **AREAEL** / **DHELEM**.

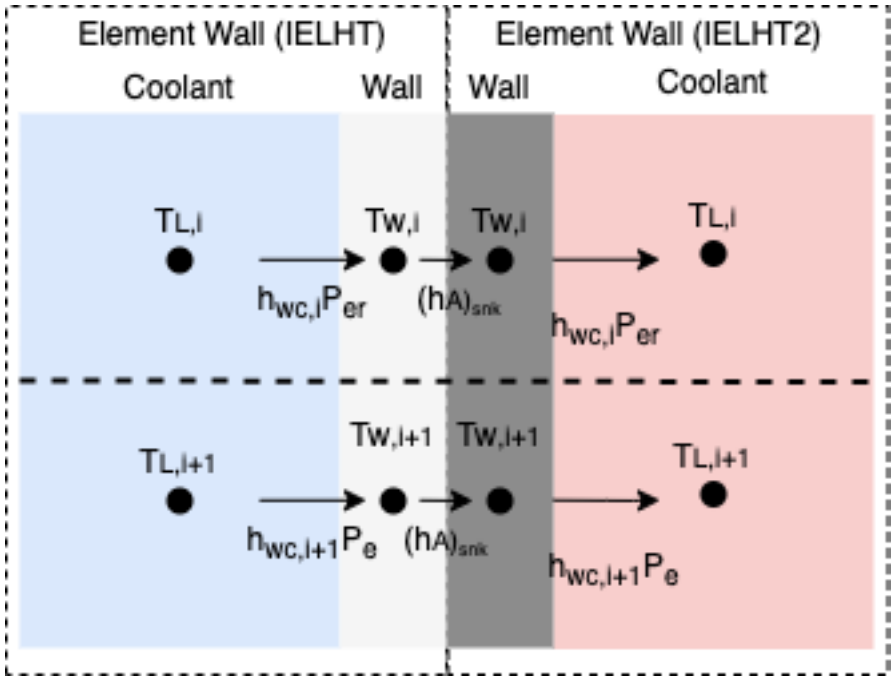


Figure 5.4.7: Component to component heat transfer between two elements.

5.4.7 RVACS/RACS Models

Three RVACS (*R*eactor *V*essel *A*uxiliary *C*ooling *S*ystem) or RACS (*R*eactor *A*ir *C*ooling *S*ystem) Models have been included in SAS4A/SASSYS-1: a simple model in which the user supplies the relevant information for the outside of the reactor vessel wall, a more detailed model in which air temperatures and flow rates are calculated by SAS, and a coupled model where an external code provides the relevant information for the region outside of the reactor vessel wall. The reactor coolant treatment and the representation of the reactor vessel wall are the same in all models. The vessel wall is represented by a combination of compressible volume walls and the walls of pipes and annular elements. The component-to-component heat transfer capability described in Section 5.4.6 above is used to remove heat from the reactor vessel walls, and the three models are used to set T_{snk}, H_{snk}, and A_{snk}, for the wall.

The input required for each of the RVACS models is described below.

5.4.7.1 Simple RVACS Model

For the simple model, the representation of the RVACS stops at the reactor vessel wall. The user supplies a heat-sink temperature and a heat transfer coefficient as a function of the reactor vessel wall temperature. For each axial node in the reactor vessel wall, the code calculates an effective heat transfer coefficient, h_{snk}, using

$$h_{\text{snk}} = S_{RV}(t)h(T_{RV}) \tag{5.4-106}$$

where

$S_{RV}(t)$ is an optional, time-dependent scaling factor,

$h(T_{RV})$ is the user-defined heat transfer coefficient (W/m^2-K),

and

T_{RV} is the reactor vessel wall temperature.

User-defined heat transfer coefficients can be defined either by providing tabulated data in **RVHTAB** and **RVHTMP** or by defining a temperature-dependent *FUNCTION* block and setting **IRVOPT** > 12.

The optional scaling factor, $S_{RV}(t)$, is defined by **IRVHTS**. If no user-defined function is provided, a default of $S_{RV}(t) = 1$ is used.

For all axial nodes, the heat-sink temperature is defined either as a constant, **TAIRVC**, or as a time-dependent function in **IDRVACSTin**.

5.4.7.2 Detailed Air Side Model

Figure 5.4.8 shows the general RVACS model and noding scheme. Wall temperature nodes are used for the reactor vessel, guard vessel, finned shell (inside and outside), outer wall, and a constant temperature deep in the concrete or the ground. Air nodes are included for the down-comer and for the up-flow section. An air inlet section and an outlet stack are also included. Either vertical or non-vertical sections of vessel wall can be treated.

The reactor vessel wall can be made up of a number of SAS4A/SASSYS-1 components, including a hot pool wall, a cold pool wall, a pipe wall, or an annular flow element wall. The annular flow element is a new type of liquid flow element that has recently been added to SAS4A/SASSYS-1, mainly for modeling the RVACS and for modeling multi-dimensional effects in pools.

Both radiative and convective heat transfer from the reactor vessel to the guard vessel are modeled. Radiation from the guard vessel to the finned shell as well as convective heat transfer to air from the guard vessel and the finned shell are treated.

The finned shell is assumed to be insulated, but less-than-perfect insulation can be treated. Temperatures on both sides of the insulation are calculated. Heat transfer from the inside of the to the outside, and from the outside of the finned shell to the incoming air, is treated. Radiation from the finned shell to be outer wall is treated as well as convective heat transfer from the outer wall to the incoming air. Also, conduction from the outer wall to a constant temperature node deep in the concrete or in the ground is handled. With perfect insulation on the finned shell and little heat transfer into the concrete, the calculation of air and wall temperatures on the down-comer side would be unnecessary; the air temperature at the bottom of the up-flow side could be set to the outside air inlet temperature. The model with heat transfer to the incoming air allows consideration of the impact on RVACS-RACS performance of limited insulation between the downcomer and the up-flow side.

As indicated in Figure 5.4.8, one axial node is included above the liquid level in the vessel. This node is included to account for heat transfer through the finned shell between incoming and outgoing air. For this node, the heat transfer coefficient between the reactor vessel and the guard vessel is set to zero.

5.4.7.2.1 Basic Equations

A. Wall Temperatures

1. Reactor Vessel

The basic equation for the reactor vessel wall temperature is

$$m_{\mathrm{w1}} C_{\mathrm{w1}} \frac{\mathrm{dT_{w1}}}{\mathrm{dt}} = h_{\mathrm{Nw1}} A_{\mathrm{Nw1}} \left(T_{\mathrm{N}} - T_{\mathrm{w1}}\right) + h_{\mathrm{w12}} A_{\mathrm{w12}} \left(T_{\mathrm{w2}} - T_{\mathrm{w1}}\right) \tag{5.4-107}$$

where

m_{w1} = reactor vessel wall mass per unit length

C_{w1} = reactor vessel specific heat

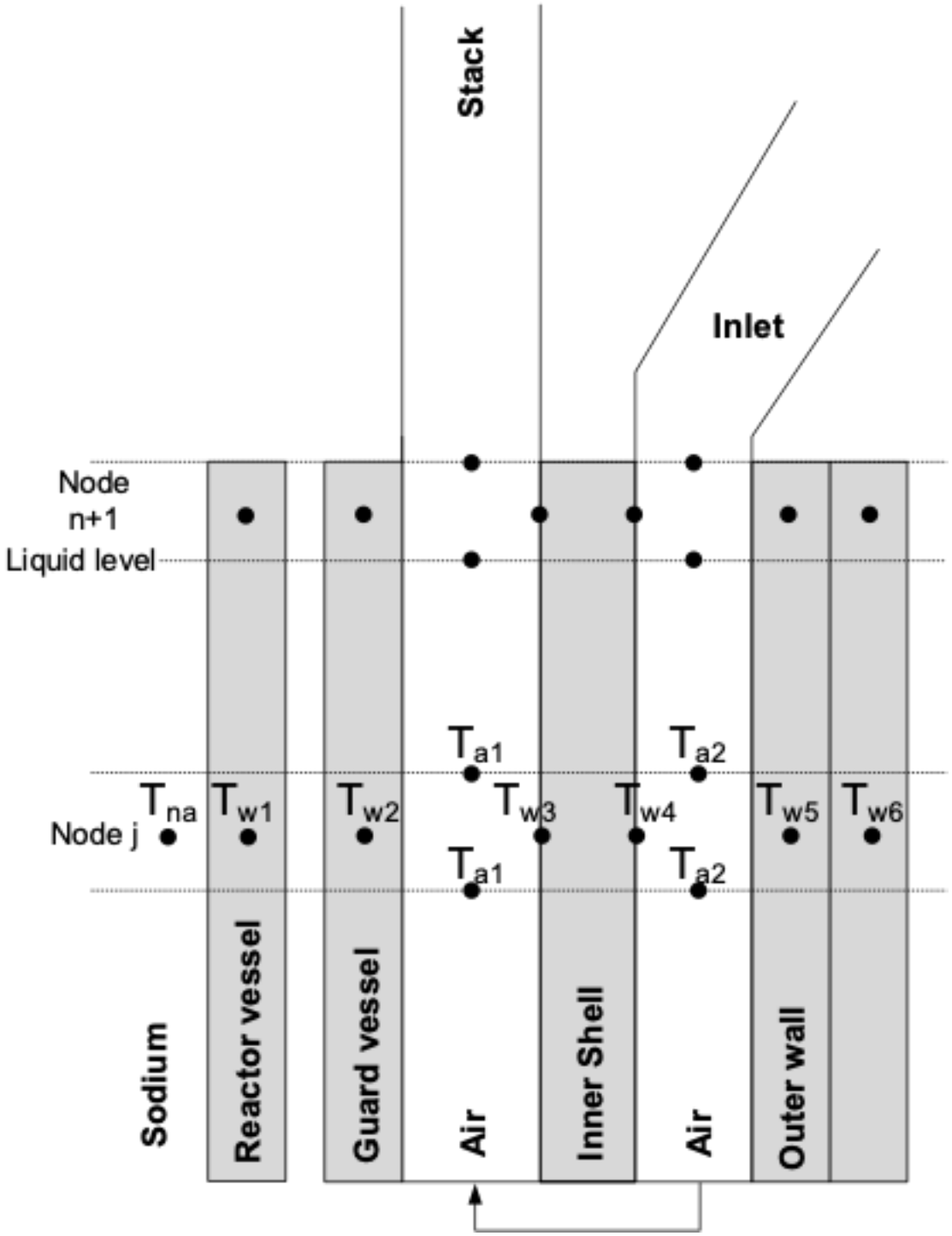


Figure 5.4.8: SAS4A/SASSYS-1 Model of an RVACS/RACS System.

T_{w1} = reactor vessel wall temperature

t = time

h_{Nw1} = heat transfer coefficient between the sodium inside the vessel and the vessel wall

A_{Nw1} = inner perimeter, or heat transfer area per unit length, of the vessel wall

T_N = sodium temperature

h_{w12} = heat transfer coefficient between the reactor vessel and the guard vessel, including both radiation and convection terms

A_{w12} = outer perimeter of the reactor vessel, **PERVAC**

T_{w12} = guard vessel temperature

2. Guard Vessel

$$\frac{(mC)_{w2}}{F_{NT}}\frac{dT_{w2}}{dt} = h_{w12}A_{w12}\left(T_{w1}-T_{w2}\right) + h_{w23}A_{w23}\left(T_{w3}-T_{w2}\right) + h_{w2a}A_{w2a}\left(T_{a1}-T_{w2}\right) \tag{5.4-108}$$

where

$(mC)_{w2}$ = heat capacity per unit length of the guard vessel, **GVMC**

F_{NT} = thermal inertia inverse scaling factor

h_{w23} = radiation heat transfer coefficient between the guard vessel and the finned shell

A_{w23} = outer perimeter of the guard vessel, **PERGVO**

T_{w3} = temperature of the finned shell inner node

T_{a1} = air temperature in the up-flow section

h_{w2a} = heat transfer coefficient between the guard vessel and air

A_{w2a} = perimeter between the guard vessel and the air, **PERGV**

F_{NT} varies depending on the stage of the simulation

$$F_{NT} = \begin{cases} \text{FINVNT} & \text{if Null Transient} \\ 1 & \text{if Transient} \end{cases} \tag{5.4-109}$$

where FINVNT is the user provided inverse scaling factor, **FINVNT**. The thermal inertia inverse scaling factor allows the detailed RVACS component to approach a steady state faster during the null transient simulation.

Setting **ILGVHT** to 1 activates an extended RVACS geometry where the heat transfer pathway in the lower-cavity under the guard vessel is taken into account. See Figure 5.4.9. Due to complex geometry that is associated with the cavity under the guard vessel, a split factor, **SPLITF**, is used to represent the ratio of the air stream thermally interacting under the guard vessel to the total air flow from the downcomer. Users are responsible for determining the split factor based on more detailed design calculations. Treating the lower-cavity wall as a single node, the base equation of the wall temperature is

$$\frac{(mC)_{LV}}{F_{NT}}\frac{dT_{LV}}{dt} = h_{RV}A_{RV}\left(T_{RV}-T_{LV}\right) + (hA)_{LV}\left(T_{aLV}-T_{LV}\right) \tag{5.4-110}$$

where

$(mC)_{LV}$ = heat capacity per unit length of the lower guard vessel, **GVMC**(1)

T_{LV} = Lower guard vessel wall temperature, T_{w2}

T_{RV} = Lower reactor vessel wall temperature, T_{w1}

h_{LV} = heat transfer coefficient between the lower guard vessel and the air in the cavity, **HTCLGV**

A_{LV} = perimeter of the lower guard vessel, **PERGV**(1)

h_{RV} = heat transfer coefficient between the reactor vessel and the guard vessel, h_{w12}

A_{RV} = perimeter of the lower reactor vessel, **PERVAC**(1)

3. Finned shell, Inner Node

$$\frac{(mC)_{w3}}{F_{NT}}\frac{dT_{w3}}{dt} = h_{w34}A_{w34}\left(T_{w4}-T_{w3}\right)+h_{w23}A_{w23}\left(T_{w2}-T_{w3}\right) + h_{w3a}A_{w3a}\left(T_{a1}-T_{w3}\right) \tag{5.4-111}$$

$(mC)_{w3}$ = heat capacity per unit length of the inside of the finned shell, **FSMCI**

T_{w3} = finned shell inner node temperature

h_{w34} = heat transfer coefficient between the inside and outside of the finned shell, **HFSRV**

A_{w34} = perimeter of the finned shell, **PERFSO**

h_{w3a} = heat transfer coefficient between the finned shell and the upflowing air

A_{w3a} = perimeter between the finned shell and the upflowing air, **PERFS**

T_{w4} = finned shell outer node temperature

4. Finned Shell, Outer Node

$$\frac{(mC)_{w4}}{F_{NT}}\frac{dT_{w4}}{dt} = h_{w45}A_{w45}\left(T_{w5}-T_{w4}\right)+h_{w34}A_{w34}\left(T_{w3}-T_{w4}\right) + h_{w4a}A_{w4a}\left(T_{a2}-T_{w4}\right) \tag{5.4-112}$$

$(mC)_{w4}$ = heat capacity per unit length of the outside of the finned shell, **FSMCO**

T_{w5} = concrete wall inner node temperature

h_{w45} = heat transfer coefficient between the finned shell and the concrete

A_{w45} = finned shell outer perimeter, **PERFSO**

T_{a2} = temperature of the down-flowing air

h_{w4a} = heat transfer coefficient between the finned shell outer surface and the down-flowing air

A_{w4a} = perimeter between the finned shell outer surface and the down-flowing air, **PERFSO**

5. Cavity Wall, Inner Node

$$\frac{(mC)_{w5}}{F_{NT}}\frac{dT_{w5}}{dt} = h_{w56}A_{w56}\left(T_{w6}-T_{w5}\right)+h_{w45}A_{w45}\left(T_{w4}-T_{w5}\right) + h_{w5a}A_{w5a}\left(T_{a2}-T_{w5}\right) \tag{5.4-113}$$

$(mC)_{w5}$ = heat capacity per unit length of the inside of the cavity wall, **CRMCI**

T_{w6} = constant temperature deep in the concrete or ground

h_{w56} = heat transfer coefficient between the cavity wall surface node and the constant temperature heat sink

A_{w56} = cavity wall perimeter, **PERFSO**

h_{w5a} = heat transfer coefficient from the cavity wall to the down-flowing air

A_{w5a} = perimeter between the down-flowing air and the cavity surface wall, **PERFSO**

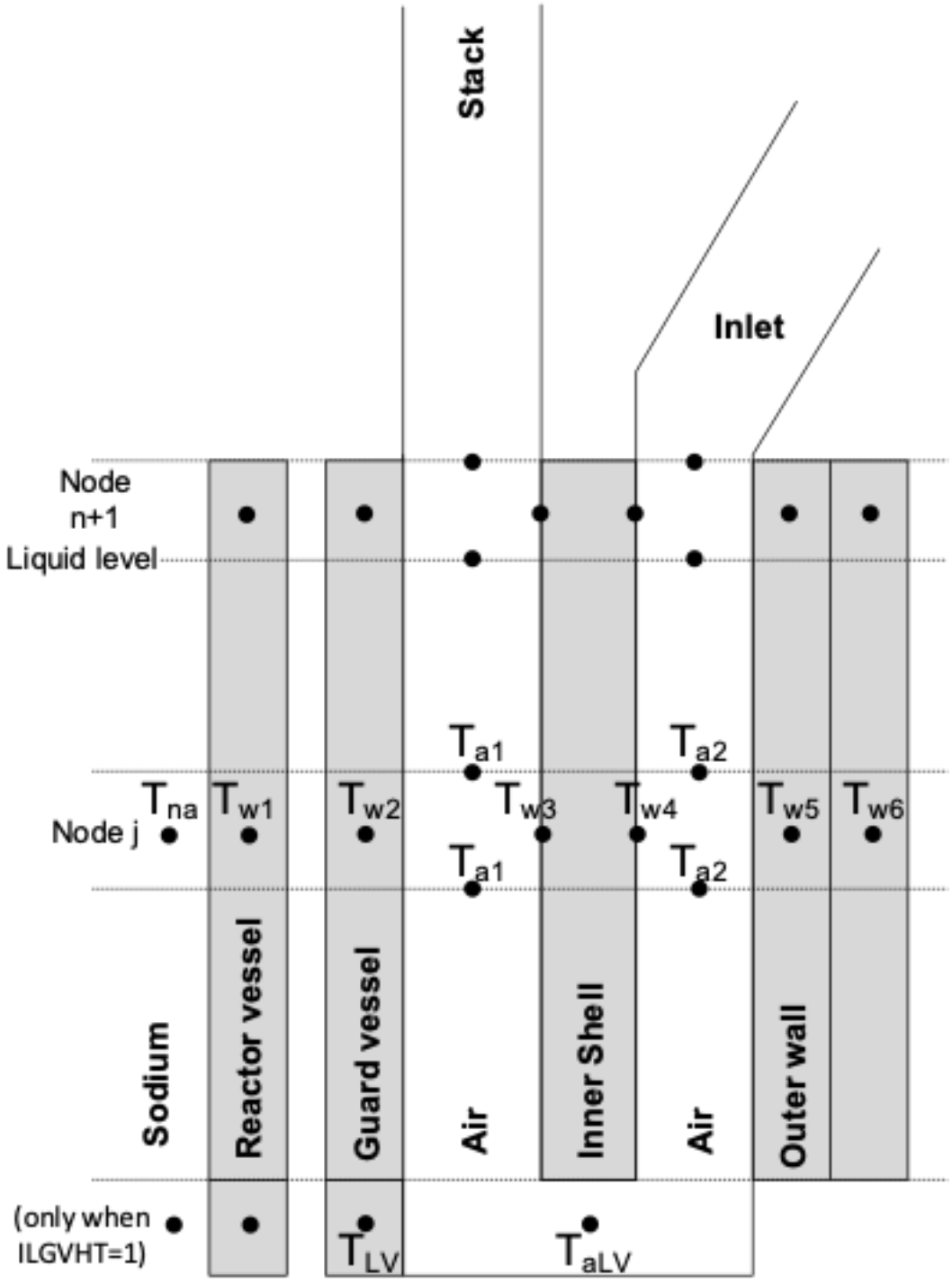


Figure 5.4.9: Lower Guard Vessel Heat Transfer Model.

B. Air Temperature

The air is treated with a quasi-static approximation, neglecting the time derivative of the air temperature and density.

1. Up-Flowing Air Between the Guard Vessel and the Finned Shell

$$w_{\mathrm{a}} C_{\mathrm{a}} \frac{\mathrm{dT_{a1}}}{\mathrm{dz}} = h_{\mathrm{wa2}} A_{\mathrm{wa2}} \left(T_{\mathrm{w2}} \left(z \right) - T_{\mathrm{a1}} \left(z \right) \right) + h_{\mathrm{wa3}} A_{\mathrm{wa3}} \left[T_{\mathrm{w3}} \left(z \right) - T_{\mathrm{a1}} \left(z \right) \right] \tag{5.4-114}$$

where

w_{a} = air mass flow rate, kg/sec.

C_{a} = air specific heat

If **ILGVHT** is 1, the riser inlet temperature is defined as a mixture of the air from the lower guard vessel and from the downcomer. Treating the air temperature in the cavity below the guard vessel as a single value, the base equation for the air temperature in the cavity is

$$(hA)_{\mathrm{LV}} \left(T_{\mathrm{LV}} - T_{\mathrm{aLV}} \right) = S w_{\mathrm{a}} C_{\mathrm{a}} \left(T_{\mathrm{aLV}} - T_{\mathrm{a2out}} \right) = w_{\mathrm{a}} C_{\mathrm{a}} \left(T_{\mathrm{a1in}} - T_{\mathrm{a2out}} \right) \tag{5.4-115}$$

where

T_{aLV} = air temperature in the cavity below the guard vessel.

S = air flow splitting factor, **SPLITF**

T_{a2out} = outlet temperature of the downflowing air between the finned shell and the concrete wall

T_{a1in} = inlet temperature of the upflowing air between the guard vessel and the finned shell

2. Down-Flowing Air Between the Finned Shell and the Concrete Wall

$$w_{\mathrm{a}} C_{\mathrm{a}} \frac{\mathrm{dT_{a2}}}{\mathrm{dz}} = h_{\mathrm{wa4}} A_{\mathrm{wa4}} \left(T_{\mathrm{w4}} \left(z \right) - T_{\mathrm{a2}} \left(z \right) \right) + h_{\mathrm{wa5}} A_{\mathrm{wa5}} \left[T_{\mathrm{w5}} \left(z \right) - T_{\mathrm{a2}} \left(z \right) \right] \tag{5.4-116}$$

C. Air Flow Rate

As shown in Figure 5.4.8, the air flow path is modeled as an inlet section, a down-flow section between the outer surface of the finned shell and the cavity wall, an up-flow section between the guard vessel and the inner surface of the finned shell, and an outlet stack. The air temperature in the inlet section is assumed to be equal to the external air temperature, which the user can define as either constant or varying with time. In the stack, the air temperature is assumed to equal the value at the outlet from the guard vessel-finned shell region. Between the inlet section and the stack, the air temperature is calculated on a node-by-node basis.

The inertia of the air is ignored, and the air flow rate is calculated by balancing the air gravity head with the loss terms.

$$\Delta p_{\mathrm{gr}} = \Delta p_{\mathrm{loss}} \tag{5.4-117}$$

The gravity head, Δp_{gr}, is calculated as

$$\begin{aligned} \Delta p_{\mathrm{gr}} = \int \rho \mathrm{gdz} = g \Bigg\{ & -\rho_{\mathrm{in}} \Delta z_{\mathrm{in}} + \rho_{\mathrm{stack}} \Delta z_{\mathrm{stack}} - \sum_{\mathrm{j}} \overline{\rho}_{\mathrm{a2}} \left(j \right) \Delta z_{\mathrm{j}} \\ & + \sum_{\mathrm{j}} \overline{\rho}_{\mathrm{a1}} \left(j \right) \Delta z_{\mathrm{j}} - \rho_{\mathrm{in}} \left(z_{\mathrm{st}} - z_{\mathrm{in}} \right) \Bigg\} \end{aligned} \tag{5.4-118}$$

where

g = acceleration of gravity, **GRAVITY**

ρ_{in} = air density in the inlet section

ρ_{stack} = air density in the stack

$\overline{\rho}_{\text{a1}}(j)$ = average air density in node j for upflow air

Δz_{j} = height of node j

Δz_{in} = elevation gain in the inlet, **XLAIRV**

Δz_{stack} = elevation gain if the stack, **XLAORV**

z_{st} = elevation of the stack outlet

and

z_{in} = inlet elevation

The loss term, Δp_{loss}, is calculated as

$$\Delta p_{\text{loss}} = \frac{w_{\text{a}}^2}{2\rho_{\text{a},1}A_{\text{a},1}^2}k_{\text{or}} + \sum_i \frac{w_{\text{a}}^2}{2\rho_{\text{a},i}A_{\text{a},i}^2} f \frac{L_i}{D_{\text{h},i}} \tag{5.4-119}$$

where

w_{a} = air flow rate

$A_{\text{a},i}$ = air flow area in node i

$\rho_{\text{a},i}$ = air density in node i

f = friction factor

k_{or} = inlet orifice coefficient

L_i = length of the node i

$D_{\text{h,i}}$ = hydraulic diameter of node i

The summation is over the inlet section (i=1), the stack, and each of the nodes between.

The friction factor is calculated as

$$f = \begin{cases} A_{\text{fr}}\text{Re}^b & \text{if Re} \geq \text{Re}_{\text{t}} \\ \frac{A_{\text{fl}}}{\text{Re}} & \text{if Re} < \text{Re}_{\text{t}} \end{cases} \tag{5.4-120}$$

where

$$\text{Re} = \text{ Reynolds number } = \frac{D_{\text{h}}w_{\text{a}}}{\mu_{\text{a}}A_{\text{a}}} \tag{5.4-121}$$

μ_{a} = viscosity

A_{fr} = user-supplied turbulent friction factor coefficient, **AFRTRV**,

b = user-supplied turbulent friction factor exponent, **BFRTRV**, and

A_{fl} = user-supplied laminar friction factor coefficient, **AFRLRV**

The value of Re_{t}, the Reynolds number for the transition from turbulent to laminar, **REYTRV**, is calculated by the code to make the friction factor continuous at the transition point.

$$A_{\text{fr}}\text{Re}_{\text{t}}^b = \frac{A_{\text{fl}}}{\text{Re}_{\text{t}}} \tag{5.4-122}$$

Or

$$\mathrm{Re}_t = \left(\frac{A_{fl}}{A_{fr}}\right)^{\frac{1}{1+b}} \tag{5.4-123}$$

D. Heat Transfer Coefficients

1. Reactor Vessel (RV) to Guard Vessel (GV)

$$\frac{1}{h_{w12}} = \frac{R_{w1}}{2} + \frac{1}{h_{cv12} + \varepsilon_{12}\sigma\left(T_{w1}+T_{w2}\right)\left(T_{w1}^2+T_{w2}^2\right)} + \frac{R_{w2}}{2} \tag{5.4-124}$$

The approximation is made that R_{w1} and R_{w2} are lumped in with h_{cv12}, so

$$h_{w12} = h_{cv12} + \varepsilon_{12}\sigma\left(T_{w1}+T_{w2}\right)\left(T_{w1}^2+T_{w2}^2\right) \tag{5.4-125}$$

where

$$\varepsilon_{12} = \frac{1}{\frac{1}{\varepsilon_{RV}} + \frac{1}{\varepsilon_{GVI}} - 1} \tag{5.4-126}$$

and

h_{cv12} = user-supplied convective heat transfer coefficient, RV to GV, **HGASRV**

ε_{RV} = emissivity of the reactor vessel wall, **EPSRV**

ε_{GVI} = emissivity of the guard vessel inner surface, **EPSGV**

σ = Stefan-Boltzmann Constant, **SIGSTB**

$$R_{w1} = G_{RV}/k_{RV} \tag{5.4-127}$$

$$R_{w2} = G_{GV}/k_{GV} \tag{5.4-128}$$

G_{RV} = thickness of the reactor vessel

G_{GV} = thickness of the guard vessel

k_{GV} = thermal conductivity of guard vessel

k_{RV} = thermal conductivity of the reactor vessel

2. GV to Finned Shell (FS)

$$\frac{1}{h_{w23}} = \frac{R_{w2}}{2} + \frac{1}{\varepsilon_{23}\sigma\left(T_{w2}+T_{w3}\right)\left(T_{w2}^2+T_{w3}^2\right)} \tag{5.4-129}$$

R_{w2} is neglected, so

$$h_{w23} = \varepsilon_{23}\sigma\left(T_{w2}+T_{w3}\right)\left(T_{w2}^2+T_{w3}^2\right) \tag{5.4-130}$$

$$\varepsilon_{23} = \frac{1}{\frac{1}{\varepsilon_{GVO}} + \frac{1}{\varepsilon_{FSI}} - 1} \tag{5.4-131}$$

ε_{GVO} = emissivity of the guard vessel outer surface, **EPSGVO**

ε_{FSI} = emissivity of the finned shell inner surface, **EPSFSI**

3. Finned Shell

h_{w34} = a constant, user-supplied, conduction coefficient, **HFSRV**

4. Outer Wall

$$\frac{1}{h_{\mathrm{w45}}} = R_{\mathrm{w5}} + \frac{1}{\varepsilon_{45}\sigma\left(T_{\mathrm{w4}} + T_{\mathrm{w5}}\right)\left(T_{\mathrm{w4}}^2 + T_{\mathrm{w5}}^2\right)} \tag{5.4-132}$$

or

$$h_{\mathrm{w45}} = \frac{\varepsilon_{45}\sigma\left(T_{\mathrm{w4}} + T_{\mathrm{w5}}\right)\left(T_{\mathrm{w4}}^2 + T_{\mathrm{w5}}^2\right)}{1 + R_{\mathrm{w5}}\varepsilon_{45}\sigma\left(T_{\mathrm{w4}} + T_{\mathrm{w5}}\right)\left(T_{\mathrm{w4}}^2 + T_{\mathrm{w5}}^2\right)} \tag{5.4-133}$$

where

$$\varepsilon_{45} = \frac{1}{\frac{1}{\varepsilon_{\mathrm{FSO}}} + \frac{1}{\varepsilon_{\mathrm{OW}}} - 1} \tag{5.4-134}$$

R_{w5} = thermal resistance in the outer wall to the location of T_{w5}, **RW5RV**

h_{w56} = a constant, user supplied, conduction coefficient, **HCONRV**

$\varepsilon_{\mathrm{FSO}}$ = emissivity of the finned shell outer surface, **EPSFS**

$\varepsilon_{\mathrm{OW}}$ = emissivity of the outer wall, **EPSOW**

5. Air

$$\frac{1}{h_{\mathrm{w2a}}} = \frac{R_{\mathrm{w2}}}{2} + \frac{1}{h_{\mathrm{w2a1}}} \tag{5.4-135}$$

again R_{w2} is neglected, so

$$h_{\mathrm{w2a}} = h_{\mathrm{w2a1}} \tag{5.4-136}$$

where the heat transfer coefficient between the air in the riser, i = 1, or downcomer, i = 2, and adjacent wall, k, is

$$h_{\mathrm{w}k\mathrm{a}i} = \frac{k_{\mathrm{ai}}}{D_{\mathrm{h,ai}}}\mathrm{Nu}_{i,k} \tag{5.4-137}$$

where

$D_{\mathrm{h,ai}}$ = hydraulic diameter for air flow

k_{ai} = thermal conductivity of the air

$\mathrm{Nu}_{i,k}$ = Nusselt number, defined as

$$\mathrm{Nu}_{i,k} = \begin{cases} C_1 {\mathrm{Re}_i}^{C_2} {\mathrm{Pr}_i}^{0.4} \left(\frac{T_{\mathrm{ai}}}{T_{\mathrm{wk}}}\right)^{C_4} & \text{if } \mathrm{Re}_i \geq \mathrm{Re}_{\mathrm{t}} \\ C_3 & \text{if } \mathrm{Re}_i < \mathrm{Re}_{\mathrm{t}} \end{cases} \tag{5.4-138}$$

$$\mathrm{Re}_i = \frac{D_{\mathrm{h,ai}} w_{\mathrm{ai}}}{A_{\mathrm{ai}}\mu_{\mathrm{ai}}} \tag{5.4-139}$$

w_{ai} = air flow rate

A_{ai} = air flow area in the riser, **AIRARV**, or downcomer **AIRAR2**

μ_{ai} = air viscosity

Pr_i = Prandtl number

Re_{t} = Reynolds number for transition from turbulent to laminar heat transfer

C_1, C_2, C_3, and C_4 are user supplied correlation coefficients, **C1RV**, **C2RV**, **C3RV**, and **C4RV**.

$C_1 \sim .023$

$C_2 \sim 0.8$

$C_3 \sim 3 - 8$

$C_4 \sim 0 - 0.4$

Also,

$$h_{\text{w3a}} = h_{\text{w3a1}} \tag{5.4-140}$$

$$h_{\text{w4a}} = h_{\text{w4a2}} \tag{5.4-141}$$

$$\frac{1}{h_{\text{w5a}}} = R_{\text{w5}} + \frac{1}{h_{\text{w5a2}}} \tag{5.4-142}$$

or

$$h_{\text{w5a}} = \frac{h_{\text{w5a2}}}{1 + R_{\text{w5}} h_{\text{w5a2}}} \tag{5.4-143}$$

E. Air Properties

Correlations used for air thermal properties are:

Density

$$\rho_{\text{a}} = .0003985 + \frac{352.25}{T} + \frac{207.89}{T^2} \ \left(\text{kg/m}^3\right) \tag{5.4-144}$$

where

T = air temperature (K)

Specific Heat

$$C_{\text{pa}} = 972.93 + .086802T + 7.8654x10^{-5}T^2 \ \text{(J/kg-K)} \tag{5.4-145}$$

Prandtl Number:

$$\text{Pr}^{0.4} = 0.89887 - 1.1906 \times 10^{-4}T + 8.6681 \times 10^{-8}T^2 \tag{5.4-146}$$

Viscosity

$$\mu = 3.872 \times 10^{-6} + 5.332 \times 10^{-8}T - 1.531 \times 10^{-11}T^2 \ \text{(kg/m-s)} \tag{5.4-147}$$

$$\mu^{0.2} = .09628 + 6.358 \times 10^{-5}T - 2.633 \times 10^{-8}T^2 \tag{5.4-148}$$

Thermal Conductivity:

$$k = 2.085 \times 10^{-3} + 8.820 \times 10^{-5}T - 2.304 \times 10^{-8}T^2 \ \text{(W/m-K)} \tag{5.4-149}$$

These correlations agree with the tables of dry air properties on page 522 of Ref. 5-6 over the range from 255.4 K (0°F) to 1088.7 K (1500°F) to within .2% for ρ_{a}, .34% for C_{pa}, .03% for $\text{Pr}^{0.4}$, 1% for μ, .4% for $\mu^{0.2}$, and 2.7% for k.

5.4.7.2.2 Finite Difference Solution

A. Finite Difference Equations

1. Reactor Vessel

The coupling between the RVACS/RACS model and the rest of SAS4A/SASSYS-1 takes place at the reactor vessel wall. Heat transfer from the vessel wall to the guard vessel is treated using the component-to-component heat transfer capability of the code. Eq. (5.4-107) becomes

$$m_{\mathrm{w1}} C_{\mathrm{w1}} \frac{\mathrm{dT_{w1}}}{\mathrm{dt}} = h_{\mathrm{Nw1}} A_{\mathrm{Nw1}} \left(T_{\mathrm{N}} - T_{\mathrm{w1}}\right) + h_{\mathrm{snk}} A_{\mathrm{snk}} \left(T_{\mathrm{snk}} - T_{\mathrm{w1}}\right) \tag{5.4-150}$$

where h_{snk}, A_{snk}, and T_{snk} are the sink heat transfer coefficient, area, and temperature used in the component-to-component heat transfer treatment. At the beginning of each time step these values are re-set as

$$h_{\mathrm{snk}} = h_{\mathrm{w12}} \frac{P_{RV_j}}{P_{component_j}} \tag{5.4-151}$$

$$A_{\mathrm{snk}} = A_{\mathrm{w12}} \Delta z_{\mathrm{j}} \tag{5.4-152}$$

$$T_{\mathrm{snk}} = T_{\mathrm{w2}} \tag{5.4-153}$$

where P_{RV_j} is the user-defined RV perimeter and $P_{component_j}$ is the perimeter of the primary system component (element or CV) which RVACS is coupled to, for node j. The vessel wall is modeled as a combination of compressible volume walls, pipe walls, and annular element walls. The thermal treatments for these components solve Eq. (5.4-150).

Determination of $P_{component_j}$ in Eq. (5.4-151) is described by Table 5.4.16, below. It should be noted that all parameters used to calculate $P_{component_j}$ are obtained directly from user input.

Table 5.4.16: Specification of $P_{component_j}$

Coupled Component	$P_{component_j}$
CV	P_{RV_j}
Pipe	$\frac{4A}{D_h}$
Annular Element	$P_{secondwall}$

2. Air Temperature

A treatment similar to the log-mean temperature difference treatment is used for the air. Fully implicit time differencing is used in the sense that the values used for T_{w2} and T_{w3} in Eq. (5.4-114) are the values at the end of the time step. Also, it is assumed that T_{w2} and T_{w3} are constant across a node. For node j, which extends from z_{j} to $z_{\mathrm{j}} + 1$, the solution of Eq. (5.4-114) then becomes

$$T_{\mathrm{a1}}\left(z'\right) = T'_{\mathrm{a1}} + \left(T_{\mathrm{a1}}\left(0\right) - T'_{\mathrm{a1}}\right) e^{-\lambda_1 z'} \tag{5.4-154}$$

where

$$z' = z - z_{\mathrm{j}} \tag{5.4-155}$$

$$T'_{\mathrm{a1}} = f_2 T_{\mathrm{w3j}} + f_3 T_{\mathrm{w3j}} \tag{5.4-156}$$

$$f_2 = \frac{h_{\mathrm{w2a}} A_{\mathrm{w2a}}}{h_{\mathrm{w2a}} A_{\mathrm{w2a}} + h_{\mathrm{w3a}} A_{\mathrm{w3a}}} \tag{5.4-157}$$

$$f_3 = 1 - f_2 \tag{5.4-158}$$

and

$$\lambda_1 = \frac{h_{\mathrm{w2a}} A_{\mathrm{w2a}} + h_{\mathrm{w3a}} A_{\mathrm{w3a}}}{w_{\mathrm{a}} C_{\mathrm{a}}} \tag{5.4-159}$$

The heat flow from the guard vessel to the air in node *j* is then

$$q_{\mathrm{2aj}} = \int_{\mathrm{z}'=0}^{\Delta z_{\mathrm{j}}} h_{\mathrm{w2a}} A_{\mathrm{w2a}} \left[T_{\mathrm{w2j}} - T_{\mathrm{a}} \left(z' \right) \right] \mathrm{dz}' \tag{5.4-160}$$

where

$$\Delta z_{\mathrm{j}} = z_{\mathrm{j+1}} - Z_{\mathrm{j}} \tag{5.4-161}$$

Combining Eq. (5.4-154) and Eq. (5.4-160) gives

$$q_{\mathrm{2aj}} = h_{\mathrm{w2a}} A_{\mathrm{w2a}} \left\{ \Delta z_{\mathrm{j}} \left(T_{\mathrm{w2j}} - T'_{\mathrm{a1}} \right) + \frac{\left[T'_{\mathrm{a1}} - T_{\mathrm{a1}} \left(0 \right) \right] \left(1 - e^{-\lambda \Delta z_{\mathrm{j}}} \right)}{\lambda_1} \right\} \tag{5.4-162}$$

The wall temperatures are solved for simultaneously with the air temperatures, so one uses

$$T_{\mathrm{w2j}} \left(t + \Delta t \right) = T_{\mathrm{w2j}} \left(t \right) + \Delta T_{\mathrm{w2j}} \tag{5.4-163}$$

and

$$T_{\mathrm{w3j}} \left(t + \Delta t \right) = T_{\mathrm{w3j}} \left(t \right) + \Delta T_{\mathrm{w3j}} \tag{5.4-164}$$

where the time step size is Δt. Then Eq. (5.4-162) has the form

$$q_{\mathrm{2aj}} = b_{\mathrm{o}} + b_2 \Delta T_{\mathrm{w2j}} + b_3 \Delta T_{\mathrm{w3j}} \tag{5.4-165}$$

where

$$b_0 = h_{\mathrm{w2a}} A_{\mathrm{w2a}} \left\{ \Delta z_{\mathrm{j}} \left[T_{\mathrm{w2j}} \left(t \right) - T'_{\mathrm{a1}} \left(t \right) \right] + \frac{\left[T'_{\mathrm{a1}} \left(t \right) - T_{\mathrm{a1}} \left(0 \right) \right]}{\lambda_1} \left(1 - e^{-\lambda_1 \Delta z_{\mathrm{j}}} \right) \right\} \tag{5.4-166}$$

$$b_2 = h_{\mathrm{w2a}} A_{\mathrm{w2a}} \left\{ \Delta z_{\mathrm{j}} \left(1 - f_2 \right) + \frac{f_2}{\lambda_1} \left(1 - e^{-\lambda_1 \Delta z_{\mathrm{j}}} \right) \right\} \tag{5.4-167}$$

and

$$b_3 = h_{\mathrm{w2a}} A_{\mathrm{w2a}} \left\{ -\Delta z_{\mathrm{j}} f_3 - \frac{f_3}{\lambda_1} \left(1 - e^{-\lambda_1 \Delta z_{\mathrm{j}}} \right) \right\} \tag{5.4-168}$$

Similarly,

$$q_{\mathrm{3aj}} = C_0 + C_2 \Delta T_{\mathrm{w2j}} + C_3 \Delta T_{\mathrm{w3j}} \tag{5.4-169}$$

$$q_{\mathrm{4aj}} = d_{\mathrm{o}} + d_4 \Delta T_{\mathrm{w4j}} + d_5 \Delta T_{\mathrm{w5j}} \tag{5.4-170}$$

and

$$q_{\mathrm{5aj}} = e_{\mathrm{o}} + e_4 \Delta T_{\mathrm{w4j}} + e_5 \Delta T_{\mathrm{w5j}} \tag{5.4-171}$$

where

$$C_0 = h_{\mathrm{w3a}} A_{\mathrm{w3a}} \left\{ \Delta z_{\mathrm{j}} \left[T_{\mathrm{w3j}} \left(t \right) - T'_{\mathrm{a1}} \left(t \right) \right] + \frac{T'_{\mathrm{a1}} \left(t \right) - T_{\mathrm{a1}} \left(0 \right)}{\lambda_1} \left(1 - e^{-\lambda_1 \Delta z_{\mathrm{j}}} \right) \right\} \tag{5.4-172}$$

$$C_2 = -h_{\mathrm{w3a}} A_{\mathrm{w3a}} \left\{ \Delta z_{\mathrm{j}} f_2 - \frac{f_2}{\lambda_1} \left(1 - e^{-\lambda_1 \Delta z_{\mathrm{j}}} \right) \right\} \tag{5.4-173}$$

$$C_3 = h_{\mathrm{w3a}} A_{\mathrm{w3a}} \left\{ \Delta z_{\mathrm{j}} \left(1 - f_3 \right) + \frac{f_3}{\lambda_1} \left(1 - e^{-\lambda_1 \Delta z_{\mathrm{j}}} \right) \right\} \tag{5.4-174}$$

$$d_{\mathrm{o}} = h_{\mathrm{w4a}} A_{\mathrm{w4a}} \left\{ \Delta z_{\mathrm{j}} \left[T_{\mathrm{w4j}} \left(t \right) - T'_{\mathrm{a2}} \left(t \right) \right] + \frac{\left[T'_{\mathrm{a2}} \left(t \right) - T_{\mathrm{a2}} \left(\Delta z \right) \right] \left(1 - e^{-\lambda_2 \Delta z_{\mathrm{j}}} \right)}{\lambda_2} \right\} \tag{5.4-175}$$

$$d_4 = h_{\mathrm{w4a}} A_{\mathrm{w4a}} \left\{ \Delta z_{\mathrm{j}} \left(1 - f_4 \right) + \frac{f_4}{\lambda_2} \left(1 - e^{-\lambda_2 \Delta z_{\mathrm{j}}} \right) \right\} \tag{5.4-176}$$

$$d_5 = -h_{\mathrm{w4a}} A_{\mathrm{w4a}} \left\{ \Delta z_{\mathrm{j}} f_5 - \frac{f_5}{\lambda_2} \left(1 - e^{-\lambda_2 \Delta z_{\mathrm{j}}} \right) \right\} \tag{5.4-177}$$

$$e_0 = h_{\mathrm{w5a}} A_{\mathrm{w5a}} \left\{ \Delta z_{\mathrm{j}} \left[T_{\mathrm{w5j}} \left(t \right) - T'_{\mathrm{a2}} \left(t \right) \right] + \frac{\left[T'_{\mathrm{a2}} \left(t \right) - T_{\mathrm{a2}} \left(\Delta z \right) \right] \left(1 - e^{-\lambda_2 \Delta z_{\mathrm{j}}} \right)}{\lambda_2} \right\} \tag{5.4-178}$$

$$e_4 = -h_{\mathrm{w5a}} A_{\mathrm{w5a}} \left\{ \Delta z_{\mathrm{j}} f_4 - \frac{f_4}{\lambda_2} \left(1 - e^{-\lambda_2 \Delta z} \right) \right\} \tag{5.4-179}$$

$$e_5 = h_{\mathrm{w5a}} A_{\mathrm{w5a}} \left\{ \Delta z_{\mathrm{j}} \left(1 - f_5 \right) + \frac{f_5}{\lambda_2} \left(1 - e^{-\lambda_2 \Delta z} \right) \right\} \tag{5.4-180}$$

$$T'_{\mathrm{a2}} = f_4 T_{\mathrm{w4j}} + f_5 T_{\mathrm{w5j}} \tag{5.4-181}$$

$$f_4 = \frac{h_{\mathrm{w4a}} A_{\mathrm{w4a}}}{h_{\mathrm{w4a}} A_{\mathrm{w4a}} + h_{\mathrm{w5a}} A_{\mathrm{w5a}}} \tag{5.4-182}$$

and

$$f_5 = 1 - f_4 \tag{5.4-183}$$

If **ILGVHT** is one, Eq. (5.4-115) for the air temperature in the cavity of the lower guard vessel becomes

$$\left(hA \right)_{\mathrm{LV}} \left(T_{\mathrm{LV}} - T_{\mathrm{aLV}} \right) = S w_{\mathrm{a}} C_{\mathrm{a}} \left(T_{\mathrm{aLV}} - T_{\mathrm{a2out}} \right) \tag{5.4-184}$$

or

$$T_{\mathrm{aLV}} = \left(1 - B_1 \right) T_{\mathrm{a2out}} + B_1 T_{\mathrm{LV}} \tag{5.4-185}$$

where

$$B_1 = \frac{\left(hA \right)_{\mathrm{LV}}}{S w_{\mathrm{a}} C_{\mathrm{a}} + \left(hA \right)_{\mathrm{LV}}} \tag{5.4-186}$$

3. Guard Vessel

Fully implicit time differencing is used for the wall temperature. Eq. (5.4-108) becomes

$$\begin{aligned} \Delta z_{\mathrm{j}} m_{\mathrm{w2}} C_{\mathrm{w2}} \frac{\Delta T_{\mathrm{w2}}}{F_{\mathrm{NT}} \Delta t} = {} & h_{\mathrm{w12}} A_{\mathrm{w12}} \left[T_{\mathrm{w1}} \left(t + \Delta t \right) - T_{\mathrm{w2}} \left(t \right) - \Delta T_{\mathrm{w2}} \right] \Delta z_{\mathrm{j}} \\ & + h_{\mathrm{w23}} A_{\mathrm{w23}} \left[T_{\mathrm{w3}} \left(t \right) + \Delta T_{\mathrm{w3}} - T_{\mathrm{w2}} \left(t \right) - \Delta T_{\mathrm{w2}} \right] \Delta z_{\mathrm{j}} - q_{\mathrm{2aj}} \end{aligned} \tag{5.4-187}$$

or

$$\alpha_2 \Delta T_{\mathrm{w2}} + \alpha_3 \Delta T_{\mathrm{w3}} = \alpha_0 \tag{5.4-188}$$

where

$$\alpha_2 = 1 + \frac{h_{w12} A_{w12} \Delta t F_{NT}}{m_{w2} C_{w2}} + \frac{h_{h23} A_{w23} \Delta t F_{NT}}{m_{w2} C_{w2}} + \frac{b_2}{\Delta z_j} \frac{\Delta t F_{NT}}{m_{w2} C_{w2}} \tag{5.4-189}$$

$$\alpha_3 = -\frac{h_{w23} A_{w23} \Delta t F_{NT}}{m_{w2} C_{w2}} + \frac{b_3 \Delta t F_{NT}}{\Delta z_j m_{w2} C_{w2}} \tag{5.4-190}$$

and

$$\alpha_0 = \frac{\Delta t F_{NT}}{m_{w2} C_{w2}} \left\{ h_{w12} A_{w12} \left[T_{w1}(t + \Delta t) - T_{w2}(t) \right] + h_{w23} A_{w23} \left[T_{w3}(t) - T_{w2}(t) \right] - \frac{b_o}{\Delta z_j} \right\} \tag{5.4-191}$$

If **ILGVHT** is one, Eq. (5.4-110) for the lower guard vessel wall becomes

$$\Delta z (mC)_{LV} \frac{\Delta T_{LV}}{F_{NT} \Delta t} = (hA)_{LV} \left(T_{aLV} - T_{LV}(t + \Delta t) \right) + h_{RV} A_{RV} \left(T_{RV}(t + \Delta t) - T_{LV}(t + \Delta t) \right) \tag{5.4-192}$$

or

$$T_{LV}(t) + \Delta T_{LV} = \frac{T_{LV}(t) + AA_1 (1 - B_1) T_{a2out} + AA_2 T_{RV} i (t + \Delta t)}{1 + AA_1 + AA_2 - AA_1 B_1} \tag{5.4-193}$$

where

$$AA_1 = \frac{(hA)_{LV}}{(mC)_{LV}} F_{NT} \Delta t \tag{5.4-194}$$

$$AA_2 = \frac{(hA)_{RV}}{(mC)_{LV}} F_{NT} \Delta t \tag{5.4-195}$$

4. Finned Shell, Inner Node

Eq. (5.4-111) becomes

$$\beta_2 \Delta T_{w2} + \beta_3 \Delta T_{w3} + \beta_4 \Delta T_{w4} = \beta_0 \tag{5.4-196}$$

where

$$\beta_0 = \frac{\Delta t F_{NT}}{m_{w3} C_{w3}} \left\{ h_{w34} A_{w34} \left[T_{w4}(t) - T_{w3}(t) \right] + h_{w23} A_{w23} \left[T_{w2}(t) - T_{w3}(t) \right] - \frac{C_o}{\Delta z_j} \right\} \tag{5.4-197}$$

$$\beta_2 = \frac{\Delta t F_{NT}}{m_{w3} C_{w3}} \left(\frac{C_2}{\Delta z_j} - h_{w23} A_{w23} \right) \tag{5.4-198}$$

$$\beta_3 = 1 + \frac{\Delta t F_{NT}}{m_{w3} C_{w3}} \left(h_{w34} A_{w34} + h_{w23} A_{w23} + \frac{C_3}{\Delta z_j} \right) \tag{5.4-199}$$

$$\beta_4 = -\frac{\Delta t h_{w34} A_{w34} F_{NT}}{m_{w3} C_{w3}} \tag{5.4-200}$$

5. Finned Shell, Outer Node

Eq. (5.4-112) becomes

$$\gamma_3 \Delta T_{w3} + \gamma_4 \Delta T_{w4} + \gamma_5 \Delta T_{w5} = \gamma_0 \tag{5.4-201}$$

where

$$\gamma_0 = \frac{\Delta t F_{NT}}{m_{w4} C_{w4}} \left\{ h_{w45} A_{w45} \left[T_{w5}(t) - T_{w4}(t) \right] + h_{m34} A_{w34} \left[T_{w3}(t) - T_{w4}(t) \right] - \frac{d_o}{\Delta z_j} \right\} \tag{5.4-202}$$

$$\gamma_3 = -\frac{\Delta t F_{NT}}{m_{w4} C_{w4}} h_{w34} A_{w34} \tag{5.4-203}$$

$$\gamma_4 = 1 + \frac{\Delta t F_{NT}}{m_{w4} C_{w4}} \left(\frac{d_4}{\Delta z_j} + h_{w34} A_{w34} + h_{w45} A_{w45} \right) \tag{5.4-204}$$

and

$$\gamma_5 = \frac{\Delta t F_{NT}}{m_{w4} C_{w4}} \left(\frac{d_5}{\Delta z_j} - h_{w45} A_{w45} \right) \tag{5.4-205}$$

6. Cavity Wall, Inner Node

Eq. (5.4-113) becomes

$$\xi_4 \Delta T_{w4} + \xi_5 \Delta T_{w5} = \xi_0 \tag{5.4-206}$$

where

$$\xi_0 = \frac{\Delta t F_{NT}}{m_{w5} C_{w5}} \left\{ h_{w45} A_{w45} \left[T_{w4}(t) - T_{w5}(t) \right] + h_{w56} A_{w56} \left[T_{w6}(t) - T_{w5}(t) \right] - \frac{e_o}{\Delta z_j} \right\} \tag{5.4-207}$$

$$\xi_4 = \frac{\Delta t F_{NT}}{m_{w5} C_{w5}} \left(\frac{e_4}{\Delta z_j} - h_{w45} A_{w45} \right) \tag{5.4-208}$$

and

$$\xi_5 = 1 + \frac{\Delta t F_{NT}}{m_{w5} C_{w5}} \left(\frac{e_5}{\Delta z_j} + h_{w45} A_{w45} + h_{w56} A_{w56} \right) \tag{5.4-209}$$

7. Cavity Wall, Outer Node

The sink temperature is assumed to be constant:

T_{w6} = constant

Solution of Finite Difference Equations

1. Simultaneous Solution of Equations

a. Down-Flowing Air, Transient Solution

The finned shell outer node and cavity wall inner node temperatures are solved for simultaneously, assuming the finned shell inner node temperature is constant. Solving Eq. (5.4-201) and Eq. (5.4-206) simultaneously gives

$$\Delta T_{w4} = \frac{\gamma_0 \xi_5 - \gamma_5 \xi_0}{\gamma_4 \xi_5 - \gamma_5 \xi_4} \tag{5.4-210}$$

Then Eq. (5.4-206) can be solved for ΔT_{w5}.

b. Up-Flowing Air, Transient Solution

The guard vessel and finned shell inner node temperatures are solved for simultaneously, assuming the finned shell outer node temperature is constant. Solving Eq. (5.4-188) and Eq. (5.4-196) simultaneously gives

$$\Delta T_{w2} = \frac{\alpha_3\beta_0 - \alpha_0\beta_3}{\alpha_3\beta_2 - \alpha_2\beta_3} \tag{5.4-211}$$

Then Eq. (5.4-188) is solved for ΔT_{w3}.

2. Solution Method

No steady-state solution for the RVACS/RACS has been coded; the initial steady-state results are obtained by running a null transient. First the air and wall temperatures are set to the air inlet temperature. Then the null transient is run to set the initial steady-state temperatures and flow rate. Finally the regular transient is run. During the null transient the core channel calculations are bypassed, the sodium flow rates are held constant, the inlet and outlet plenum temperatures are held constant, and temperatures and air flow rates are calculated for the rest of the sodium and the RVACS/RACS. The routines used in the null transient are the same as those used in the regular transient.

An iteration on air flow rate is used in the transient solution. An air flow rate is assumed, temperatures are calculated for this air flow rate, the air gravity head is calculated, and the air pressure loss is calculated. Then the gravity head is compared with the pressure loss; and if the two do not balance, another air flow rate is tried. The iteration on air flow rate continues until a balance is achieved. The temperature calculation starts at the inlet and works down the inlet side then back up the air up-flow side. The heat transfer coefficients and the heat transfer across the finned shell are calculated based on conditions at the beginning of the time step. The rest of the calculation is fully implicit in its time differencing.

5.4.7.3 Coupled RVACS Model

For the coupled model, the representation of the RVACS stops at the Reactor Vessel (RV) wall. At the end of each PRIMAR-4 time step, SAS4A/SASSYS-1 transfers the temperate of each axial node in the RV wall to an external code. The external code is expected to provide a sink temperature, T_{snk}, and a heat transfer coefficient, h_{snk} for each node. T_{snk} and h_{snk} are assumed to remain constant during the subsequent PRIMAR-4 time step, after which it will be updated again.

Due to the uniqueness of the SAS4A/SASSYS-1 null transient, the coupled RVACS model has been developed such that it can:

- Participate in the null transient, **IRVOPT** = -1,
- Not participate in the null transient, **IRVOPT** = -1000,
- Call the simple RVACS model during the null transient, **IRVOPT** < -1000.

If a user selects for the coupled RVACS model to participate in the null transient, the external code must be capable of handling the reset of the simulation elapsed time. The simulation elapsed time will reset when SAS4A/SASSYS-1 transitions from a null transient to the first transient calculation step.

Note: When **IRVOPT** < -1000, input for the simple RVACS model must be provided to support the null transient calculation.

The coupled RVACS model relies on ZMQ for data transfer between SAS4A/SASSYS-1 and the external code. ZMQ is a C++ asynchronous messaging library that allows for data transfer interfaces that are simple to create and flexible *[5-9]*. At the beginning of a SAS4A/SASSYS-1 calculation containing a coupled RVACS model, SAS will create a `ZMQ_PAIR` socket bound to a random (ephemeral) TCP port. If the binding is successful,

the RVACS coupling service is established, and a message is written to the SAS4A/SASSYS-1 log file. If the binding fails, an error is reported in the SAS4A/SASSYS-1 output file.

To determine the RVACS service port, an external code must first connect to the SAS "Harbor Master" on port 60439 using a `ZMQ_REQ` socket and send a lookup request to determine which port the RVACS service is bound to. This process typically occurs at the start of the null transient, or in the case of a restart, at the start of the first transient calculation. Once the port is identified, the external code must connect to that port using a `ZMQ_PAIR` socket for RVACS communications.

At the end of each PRIMAR-4 time step SAS4A/SASSYS-1 will send a multi-part ZMQ message containing the following:

Table 5.4.17: ZMQ Message Parts Sent to the External Code.

Variable	Description	Type
`nMesh`	The number of nodes representing the RV wall.	`c_int`
`isNull`	A flag indicating if SAS4A/SASSYS-1 is computing a null transient step, `isNull = 1` or a regular transient, `isNull = 0`.	`c_int`
`isSaved`	A flag indicating if SAS4A/SASSYS-1 saved a restart after the previous time step.	`c_int`
`t`	The time at the start of the PRIMAR-4 time step (seconds).	`c_double`
`dt`	The size of the PRIMAR4 time step (seconds).	`c_double`
`dz(nMesh)`	An array of size `nMesh` containing the height of each RV wall node (m).	`c_double`
`T1(nMesh)`	An array of size `nMesh` containing the temperature of each RV wall node (K).	`c_double`

SAS4A/SASSYS-1 then reads in a data package containing:

Table 5.4.18: ZMQ Message Parts Received from the External Code.

Variable	Description	Type
`nMesh`	The number of nodes representing the RV wall.	`c_int`
`T2(nMesh)`	An array of size `nMesh` containing T_{snk} for each RV wall node (K).	`c_double`
`H12(nMesh)`	An array of size `nMesh` containing h_{snk} for each RV wall node (W/m^2-K).	`c_double`

SAS4A/SASSYS-1 will wait at the read stage until the external code has sent a response. If the external code provides data of the wrong shape, i.e. the values of `nMesh` are not consistent, or sends an incomplete response, SAS4A/SASSYS-1 will print an error message and exit.

An example Coupled RVACS routine is shown below. The example returns a constant heat sink temperature and heat transfer coefficient for each RV node until the time reaches 100 seconds.

```
#include <zmq.h> // See https://github.com/zeromq/libzmq
#include <iostream>
#include <string.h>
#include <assert.h>
#include <unistd.h>

struct Service{
```

(continues on next page)

(continued from previous page)

```
        char tag[4];
        int  id;
};

int main (void)
{
        // Set up scalar variables
        int nMesh, isNull, rc, port;
        int saved;
        double time, dt;
        Service  serv, reply;
        char buff[25];

        printf ("Connecting to SAS…\n");
        void *context = zmq_ctx_new(); // Create a new context
        void *request = zmq_socket(context, ZMQ_REQ);  // Create a socket on␣
↪that context
        zmq_connect(request, "tcp://localhost:60439"); // Connect the socket to␣
↪TCP:60439

        // Ask SAS what port RVACS is communicating on
        std::strncpy(serv.tag,"RVAC",4);
        serv.id = 0;

        for(;;) {
                // Ask to connect to RVACS
                assert(zmq_send(request, &serv, sizeof(serv), 0) > 0);
                printf("Sent\n");
                // Recieve the reply from the harbor
                assert(zmq_recv(request, &reply, sizeof(reply), 0) > 0);
                printf("Recieved\n");
                if (strncmp(reply.tag,"PORT",4) == 0) {
                        port = reply.id; // If a port was revieved set the port␣
↪number
                        break;
                } else if (strncmp(reply.tag,"WAIT",4) == 0) {
                        usleep(1 * 1000);
                        // usleep takes sleep time in us (1 millionth of a␣
↪second)
                } else {
                        return -1;
                }
        }

        zmq_close(request);
        assert(port > 0);
        snprintf(buff,25,"tcp://localhost:%d",port);
        request = zmq_socket(context, ZMQ_PAIR); // Create a socket on that␣
↪context
        zmq_connect(request, buff); // Connect the socket to the new port
        printf("Connected\n");

        // Receive the number of nodes on the RVACS wall
        assert(zmq_recv(request, &nMesh, sizeof(int), 0) > 0);
        assert(nMesh > 0);
```

(continues on next page)

(continued from previous page)

```
        double Z[nMesh], T1[nMesh], T2[nMesh], h12[nMesh];
        // Receive the null transient information
        assert(zmq_recv(request, &isNull, sizeof(int), 0) > 0);
        // Receive the save flag
        assert(zmq_recv(request, &saved, sizeof(int), 0) > 0);
        // Receive the start time
        assert(zmq_recv(request, &time, sizeof(double), 0) > 0);
        // Receive the timestep
        assert(zmq_recv(request, &dt, sizeof(double), 0) > 0);
        // Receive the axial mesh
        assert(zmq_recv(request, &Z, sizeof(Z), 0) > 0);
        // Receive the RV temperature
        assert(zmq_recv(request, &T1, sizeof(T1), 0) > 0);
        // Set the Sink temperature and heat transfer coefficients to
        // constant values
        for ( int i = 0; i < nMesh; i++ ) {
                T2[i] = 560.0;
                h12[i] = 166.953;
        }
        // Send back the number of incoming nodes
        assert(zmq_send(request, &nMesh, sizeof(int), ZMQ_SNDMORE) > 0);
        // Send back the sink temperature
        assert(zmq_send(request, &T2, sizeof(T2), ZMQ_SNDMORE) > 0);
        // Send back the sink heat transfer coefficient
        assert(zmq_send(request, &h12, sizeof(h12), 0) > 0);

        // Continue communicating with SAS until the SAS time reaches 100.
        while(time+dt<100.-1E-6) {
                assert(zmq_recv(request, &nMesh, sizeof(int), 0) > 0);
                assert(zmq_recv(request, &isNull, sizeof(int), 0) > 0);
                assert(zmq_recv(request, &saved, sizeof(int), 0) > 0);
                assert(zmq_recv(request, &time, sizeof(double), 0) > 0);
                assert(zmq_recv(request, &dt, sizeof(double), 0) > 0);
                assert(zmq_recv(request, &Z, sizeof(Z), 0) > 0);
                assert(zmq_recv(request, &T1, sizeof(T1), 0) > 0);

                assert(zmq_send(request, &nMesh, sizeof(int), ZMQ_SNDMORE) > 0);
                assert(zmq_send(request, &T2, sizeof(T2), ZMQ_SNDMORE) > 0);
                assert(zmq_send(request, &h12, sizeof(h12), 0) > 0);

                printf ("Time: %f\n",time+dt);
        };

        // Close the socket
        zmq_close(request);
        // Destroy the context
        zmq_ctx_destroy(context);
        printf ("Done!\n");
        return 0;
}
```

5.4.7.4 RVACS Input Description

Table 5.4.19: Input in INPMR4 (Block 3) for all RVACS Models

Variable	Description
NSCRVC	Number of sections in the RVACS model.
IRVOPT	RVACS model selection
IELRVC	Element number or -ICV, starting at the bottom and going up.
NANRVC	Number of nodes in this section.
IDBRV	Debug parameter for RVACS

Table 5.4.20: Input in INPMR4 (Block 3) for the simple RVACS Model

Variable	Description
IDRVACSTin	*FUNCTION* ID for RVACS heat-sink temperature as a function of time. If **IDRVACSTin** = 0, the heat-sink temperature is specified by **TAIRVC**.
IRVHTS	*FUNCTION* ID for the RVACS heat transfer coefficient scaling factor as a function of time. If **IRVHTS** = 0, the scaling factor defaults to one.

Table 5.4.21: Input in INPMR4 (Block 3) for the detailed RVACS Model

Variable	Description
IDRVACSTin	*FUNCTION* ID for RVACS air inlet temperature as a function of time. If **IDRVACSTin** = 0, **TAIRVC** will be used.
IDRVACSKin	*FUNCTION* ID for RVACS air inlet orifice coefficient as a function of time. If **IDRVACSKin** = 0, **ORFIN** will be used.
ILGVHT	Option to specify whether the first element in the RVACS model will be treated as a lower cavity mixing region.

Table 5.4.22: Input in PMR4IN (Block 18) for all RVACS Models

Variable	Range	Description
XLRVC	>=0.0	Length of RVACS section.
SLRVC	>=0.0 <=1.0	Slope of RVACS section.
PERVAC	>0.0	Reactor vessel perimeter.
ZBRVC	-	The bottom elevation of RVACS.

Table 5.4.23: Input in PMR4IN (Block 18) for the simple RVACS Model

Variable	Range	Description
RVHTAB	!=0.0	Table of H vs T.
RVHTMP	!=0.0	Reactor vessel temperature for RVHTAB table.
TAIRVC	!=0.0	Air inlet temperature. Ignored if **IDRVACSTin** > 0.

Table 5.4.24: Input in PMR4IN (Block 18) for the detailed RVACS Model

Variable	Range	Description
PERGV	>0.0	Perimeter of the guard vessel for convective heat transfer.
PERGVO	>0.0	Perimeter of the guard vessel for radiative heat transfer.
PERFS	>0.0	Perimeter of the inner finned shell for convective heat transfer.
PERFSO	>0.0	Perimeter used for 1) conduction between the inner finned shell and outer finned shell, 2) convection from the outer finned shell to the air, 3) convection from the outer wall to the air, 4) radiation from the outer finned shell to the outer wall, and 5) conduction from the outer wall inner node to the outer wall outer node.
AIRARV	>0.0	Flow area between the guard vessel and finned shell.
AIRAR2	>0.0	Flow area between the finned shell and the outer wall
EPSRV	>=0.0	Emissivity of the reactor vessel.
EPSGV	>=0.0	Emissivity of the inner guard vessel wall.
EPSGVO	>=0.0	Emissivity of the outer guard vessel wall.
EPSFSI	>=0.0	Emissivity of the inner finned shell wall.
EPSFS	>=0.0	Emissivity of the outer finned shell wall.
EPSOW	>=0.0	Emissivity of the outer wall.
GVMC	>0.0	Heat capacity of the guard vessel wall per unit length.
FSMCI	>0.0	Heat capacity of the finned shell inner node wall per unit length.
FSMCO	>0.0	Heat capacity of the finned shell outer node wall per unit length.
CRMCI	>0.0	Heat capacity of the outer wall inner node wall per unit length.
CRMCO	>0.0	Heat capacity of the outer wall outer node wall per unit length.
HGASRV	>=0.0	Heat transfer coefficient between the reactor vessel and guard vessel. Can include convective and conductive heat transfer.
HFSRV	>=0.0	Heat transfer coefficient between the inner and outer finned shell nodes.
RW5RV	>0.0	Thermal resistance of the outer wall inner node.
HCONRV	>=0.0	Heat transfer coefficient of the outer wall outer node.
TW6RV	>0.0	Temperature of the outer wall outer node.
TAIRVC	>0.0	Air inlet temperature. Ignored if IDRVACSTin > 0.
XLAIRV	>0.0	Length of the inlet stack.
DHAIRV	>0.0	Hydraulic diameter of the inlet stack.
AARIRV	>0.0	Flow area of the inlet stack.
XLAORV	>0.0	Length of the outlet stack.
DHAORV	>0.0	Hydraulic diameter of the outlet stack.
AARORV	>0.0	Flow area of the outlet stack.
C1RV	>0.0	First coefficient of the heat transfer correlation for air.
C2RV	-	Second coefficient of the heat transfer correlation for air.
C3RV	>0.0	Third coefficient of the heat transfer correlation for air.
C4RV	-	Fourth coefficient of the heat transfer correlation for air.
REYTRV	>=0.0	Reynolds number at which laminar flow transitions to turbulent flow. This input is overwritten according to Eq. (5.4-123)
SIGSTB	>=0.0	Stefan-Boltzmann constant.
AFRTRV	>=0.0	First turbulent friction factor parameter.
BFRTRV	-	Second turbulent friction factor parameter.
AFRLRV	>=0.0	Laminar friction factor parameter.

continues on next page

Table 5.4.24 – continued from previous page

Variable	Range	Description
ORFIN	>=0.0	Inlet orifice coefficient. Ignored if **IDRVACSKin** > 0.
XLUNRV	>0.0	Length of upper reactor vessel node.
UNGVMC	>0.0	Heat capacity of the upper guard vessel node per unit length.
UNFSMC	>0.0	Heat capacity of the upper finned shell node per unit length.
AIRAUN	>0.0	Flow area of the upper node.
FINVNT	>=0.0	Inverse scaling factor for structure thermal inertia during null transient. Default: 10.0.
SPLITF	>=0.0 <=1.0	Splitting factor for the air flow if the lower cavity mixing region is used.
HTCLGV	>=0.0	Heat transfer coefficient between the lower guard vessel and the air in the lower cavity.

5.4.8 Stratified Volume Model

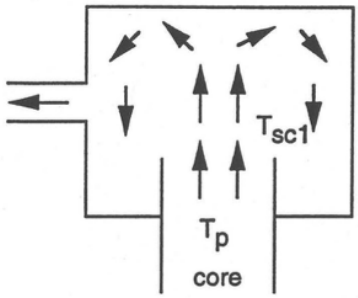
In addition to the uniform mixing compressible volume model described in Section 5.4.4, PRIMAR-4 contains a stratified temperature model for the liquid in a compressible volume. This stratified model can be used for an outlet plenum and/or for a pool. This model borrows from the PLENUM-2A model [5-7] of Howard and Lorenz, but the PRIMAR-4 model has been extended beyond the capabilities of the PLENUM-2A model. Borrowed from PLENUM-2A is the concept of a small number of distinct temperature regions in the coolant, separated by horizontal interfaces. Also, borrowed are the concept of distinct stages in the calculation, a plume height correlation, and a correlation for interface rise due to entrainment of a hot layer into a cooler plume rising from the core outlet. One extension of the PRIMAR-4 model is the provision for handling up transients as well as down transients: PLENUM-2A will only handle transients in which the core outlet temperature is cooler than the plenum temperature, whereas the PRIMAR-4 model will also handle transients in which the core outlet temperature is hotter than the plenum temperature. Another extension is the option to handle a horizontal discharge from an IHX into a cold pool: PLENUM-2A will only handle a vertical discharge from the core into an outlet plenum. The code handles up to three regions and five stages, whereas PLENUM-2A considers only two regions and three stages. Also, the PRIMAR-4 model treats thermal conduction across the interface between regions, and this model includes detailed multi-node wall temperature calculations.

Figure 5.4.10 and Figure 5.4.11 shows the various stages and cases considered in this stratified model. At the start of a transient in which the core outlet temperature is dropping, the plume in the outlet plenum goes to the top of the plenum; and the outlet plenum is fully mixed, giving stage 1. As the core outlet temperature and velocity drop, the plume no longer reaches the top of the plenum. This leads to the start of stage 2 in which the outlet coolant goes to layer 1. In stage 2 the layer boundary is at the elevation of the core outlet. After enough cool liquid has entered layer 1 to fill one quarter of its volume, stage 3, case 3.1 begins. In this case, the plume coolant still goes to layer 1, but the interface between layers rises as liquid is added to layer 1. In this case, the plume also entrains hot liquid from the interface into layer 1. If the core outlet temperature at the start of the transient becomes hotter than the outlet plenum temperature, then stage 3, case 3.2 is entered. In this case, the core outlet coolant goes to a top hot layer, entraining cool outlet plenum liquid as it passes through. The three layer cases of stages 4 and 5 can occur in the later stages of a transient if the core outlet temperature starts out rising and later falls, or if the core outlet temperature starts out falling and later rises. If the coolant inlet into the volume is horizontal, as in the discharge of an IHX into a cold pool, then only stages 1, 3, and 5 are used.

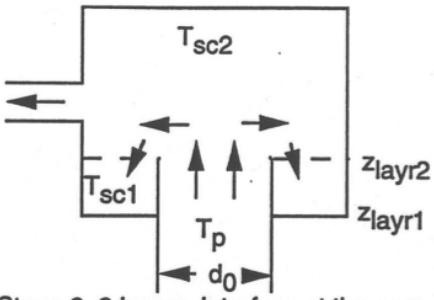
The jet height or plume height is calculated from an equation given by Yang [5-8]:

$$h_{\text{jet}} = 1.0484\text{Fr}^{0.785} \tag{5.4-212}$$

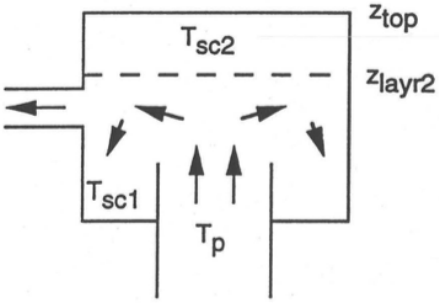
where



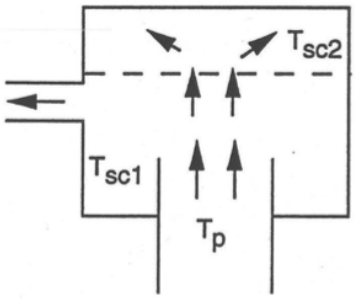
Stage 1, fully mixed, 1 layer



Stage 2, 2 layers, interface at the core outlet, filling the bottom with cool liquid



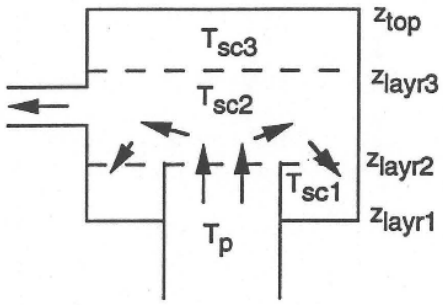
Stage 3, case3.1, 2 layers, interface moving $T_p < (T_{sc1} + T_{sc2})/2$, entrainment at the interface, plume height < z_{top}



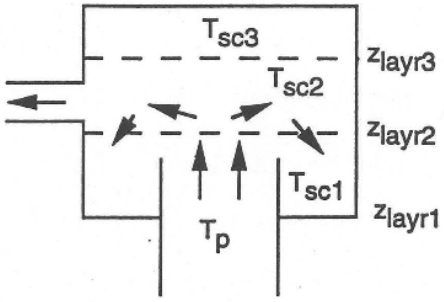
Stage 3, case 3.2, 2 layers, interface moving, $T_p > (T_{sc1} + T_{sc2})/2$, hot outlet coolant goes to upper layer, entrains from the lower layer as it passes through

T_p = plume temperature = core outlet temperature
T_{sci} = temperature in layer i
z_{top} = elevation at top of plenum
z_{layri} = interface elevation at bottom of layer i

Figure 5.4.10: Stratified Volume Stages.

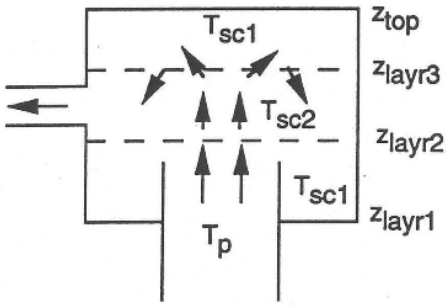


Stage 4, 3 layers, interface at the core outlet, filling the bottom with cool liquid, plume height < z_{layr3}



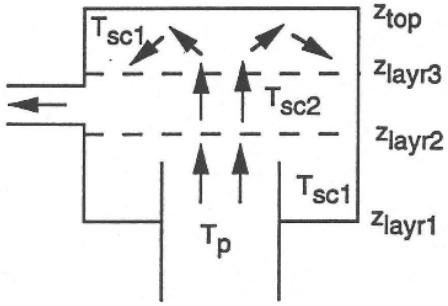
Stage 5, case 5.1, 3 layers, interfaces moving, cool plume liquid goes to layer 1 entrainment from layer 2 at z_{layr2}

$T_p < (T_{sc1} + T_{sc2})/2$



Strage 5, case 5.2, 3 layers, interfaces moving, plume passes through layer 1 to layer 2, entraining from layer 1 as it passes through, also entraining from layer 3 at z_{layr3}

$T_{sc2} + T_{sc3} > 2T_p > T_{sc1} + T_{sc2}$



Stage 5, case 5.3, 3 layers, interfaces moving, plume passes through layers 1 and 2, goes to layer 3, entraining from layers 1 and 2 as it passes through

$T_p > (T_{sc2} + T_{sc3})/2$

Figure 5.4.11: Stratified Volume Stages (Cont'd).

h_{jet} = height of the jet or the plane

$$\text{Fr} = \text{ Froude number } = \frac{v_{\text{o}}^2 \rho_{\text{plume}}}{g\, r_o^2 \left(\rho_{\text{plume}} - \rho_{\text{plenum}}\right)} \tag{5.4-213}$$

v_{o} = core exit velocity

ρ_{plume} = density of the plume

ρ_{plenum} = density of the plenum

r_{o} = core effective radius

g = acceleration of gravity

For entrainment at an interface, Howard and Lorenz give

$$w_{\text{ent}} = 0.2\pi \rho_{\text{plume}} V_{\text{j}} d_{\text{j}}^2 F_{\text{f}}^{-1.1} \tag{5.4-214}$$

where

v_{j} = plume average velocity at the interface and

d_{j} = plume effective diameter at the interface

w_{ent} = entrainment rate (kg/s).

The values of v_{j} and d_{j} depend on elevation and on whether the interface occurs within the zone of flow establishment or in the zone of established flow. The elevation change, z_{o}, from the core outlet to the top of the zone of flow establishment is

$$z_o = \frac{r_o}{.111} \tag{5.4-215}$$

For $z < z_{\text{o}}$, or the zone of flow establishment:

$$\frac{v_{\text{j}}}{v_{\text{o}}} = \frac{.25 + .02095\left(\frac{z}{d_{\text{o}}}\right) + .003969\left(\frac{z}{d_{\text{o}}}\right)^2}{\left[\frac{1}{2} + .1052\left(\frac{z}{d_{\text{o}}}\right)\right]^2} \tag{5.4-216}$$

and

$$\frac{d_{\text{j}}}{d_{\text{o}}} = 1 + .2104\left(\frac{z}{d_{\text{o}}}\right) \tag{5.4-217}$$

For $z > z_{\text{o}}$, or the zone of established flow,

$$\frac{v_{\text{j}}}{v_{\text{o}}} = \frac{2.018}{\frac{z}{d_{\text{o}}}} \tag{5.4-218}$$

and

$$\frac{d_{\text{j}}}{d_{\text{o}}} = .8649\left(\frac{z}{d_{\text{o}}}\right) \tag{5.4-219}$$

For the wall temperatures, multi-node treatments are used. The vertical wall around the outside of the outlet plenum or pool is treated with a number of vertical nodes. Each vertical node contains a number of lateral nodes, with coolant in contact with the first node. There is also an option to have another coolant compressible volume in contact with the last lateral node to account for heat transfer from a hot outlet plenum to a cold pool. The model has an option for a horizontal wall at the top or bottom of the plenum. This wall is handled with a 1-D multinode treatment. Again, the first node is in contact with the plenum liquid, and the last node can be in contact with the coolant in another compressible volume.

The input for this model is as follows:

Table 5.4.25: Input for the Stratified Volume Model

Block	Location	Name	Meaning
3	1313	NSTRCV	Number of stratified compressible volumes
3	1314-1316	ICVSTR(ICVST)	1 for vertical coolant inlet, as in an outlet plenum 2 for a horizontal coolant inlet
3	1320-1322	NUMWAL(ICVST)	Number of wall sections
3	1323-1325	IFSTWL(ICVST)	Wall number of the first wall section
3	1326-1334	IWLHRZ(IW)	0 for a vertical wall 1 for a horizontal wall at the top of a CV 2 for a horizontal wall at the bottom of a CV
3	1335-1343	NVNDWL(IW)	Number of vertical nodes in a vertical wall. NVNDWL = 1 for a horizontal wall
3	1344-1352	NLNDWL(IW)	Number of lateral nodes in a wall section, Max. = 8 Note: sum(NVNDWL*NLNDWL) <= 300.
3	1353-1361	ICV2WL(IW)	Number of the CV in contact with the outer side of the wall section. = 0 for an adiabatic outer boundary. If ICV2WL > 38, ICV2WL = the temperature of a constant temperature heat sink.
3	1362	IDBSTR	Debug flag for stratified temperature model = 0, no debug prints = 1, final results only = 5, everything
3	1363	ISTDBS	PRIMAR time step when stratified debug starts.
3	1364	ISTSTP	Stop the run at PRIMAR step ISTSTP. Not used if ISTSTP = 0 or NSTRCV = 0.
3	1365	IFT16	Write out stratified CV output to STRATCV.dat every IFT16 PRIMAR steps. No output if IFT16 = 0.
18	5008	RCORE	Core radius for use in Froude number.
18	5009-5017	HCSTWL(IW)	Coolant heat transfer coefficient at the inner surface of the wall section.
18	5027-5035	ASTWL(IW)	Area of the wall section
18	5036-5107	HINVWL(I,IW)	Thickness/thermal conductivity of node I in the wall section I = 1 - 8
18	5108-5179	XMCSTW(I,IW)	Mass x heat capacity of node I in the wall.
18	5180-5182	ZINST(ICVST)	z of inlet, used only for a vertical inlet. Otherwise ZOUTEL(IELL) is used.
18	5183-5185	VOLBLI(ICVST)	Volume below the inlet, not used if there is a cover gas in the CV. In this case, the information is obtained from other input for the CV.
18	5186-5188	EPSTST(ICVST)	Minimum temperature difference for switching stages.
18	5189-5191	XLENTR(ICVST)	Entrainment length. A hot plume with a flow rate Wh, rising through a cool layer of thickness dz, will entrain cool liquid at a rate (dz/XLENTR) x Wh.

5.5 Steam Generator

In addition to the simple table look-up steam generator model described in Section 5.4.5, SAS4A/SASSYS-1 contains a detailed steam generator model. The simple table look-up model is intended for cases in which the steam generator behavior has little impact on the results of a transient. The detailed model is intended for cases in which changes in the steam generator heat removal rate and/or gravity head have a significant impact on the results. Also, the detailed steam generator is required when the balance of plant model is used. The detailed steam generator model is described in Chapter 7.

5.6 Air Dump Heat Exchanger

SAS4A/SASSYS-1 contains a simple quasi-static sodium-to-air heat exchanger model. This model is intended mainly for use with a DRACS (*D*irect *R*eactor *A*uxiliary *C*ooling *S*ystem), as shown in Figure 5.6.1; but it could also be used as a main heat rejection component, such as for the air dump heat exchangers on the FFTF reactor.

The model can handle either forced convection or natural circulation air flow. For a forced convection case, the user supplies the air flow as a function of time. For a natural circulation case, the user supplies a table of the stack inlet loss coefficient vs. time. The stack inlet loss coefficient represents the shutter opening, which may change during the transient.

A log mean temperature model is used for both the sodium side and the air side of the heat exchanger. Heat capacity of the tube structural material is ignored.

5.6.1 Methodology

Equations used in the model are as follow: For the coolant, we ignore storage of energy within the heat exchanger and assume specific heat is independent of temperature so that an energy balance gives

$$Q = w_{\mathrm{Na}} C_{\mathrm{Na}} \left(T_{\mathrm{Nat}} - T_{\mathrm{Nab}}\right) = w_{\mathrm{a}} C_{\mathrm{a}} \left(T_{\mathrm{at}} - T_{\mathrm{ab}}\right) \ , \tag{5.6-1}$$

where

Q = heat removal rate,

w = mass flowrate,

C = specific heat,

T = temperature,

a = air side subscript,

Na = liquid metal side subscript,

t = subscript for the top of the heat exchanger,

b = subscript for the bottom of the heat exchanger.

The momentum equation for the air side is from Section 5.16.4

$$w^2 = \frac{\left(\rho_{\mathrm{c}} - \rho_{\mathrm{h}}\right) g \updownarrow A_{\mathrm{R}}^2}{\frac{K_{\mathrm{SI}}}{2\rho_{\mathrm{c}}}\left(\frac{A_{\mathrm{g}}}{A_{\mathrm{SI}}}\right)^2 + \frac{K_{\mathrm{HX}}}{2\rho_{\mathrm{c}}}\left(\frac{A_{\mathrm{g}}}{A_{\mathrm{HX}}}\right)^2 + \frac{K_{\mathrm{SO}}+K_{\mathrm{R}}}{2\rho_{\mathrm{h}}}} \tag{5.6-2}$$

where

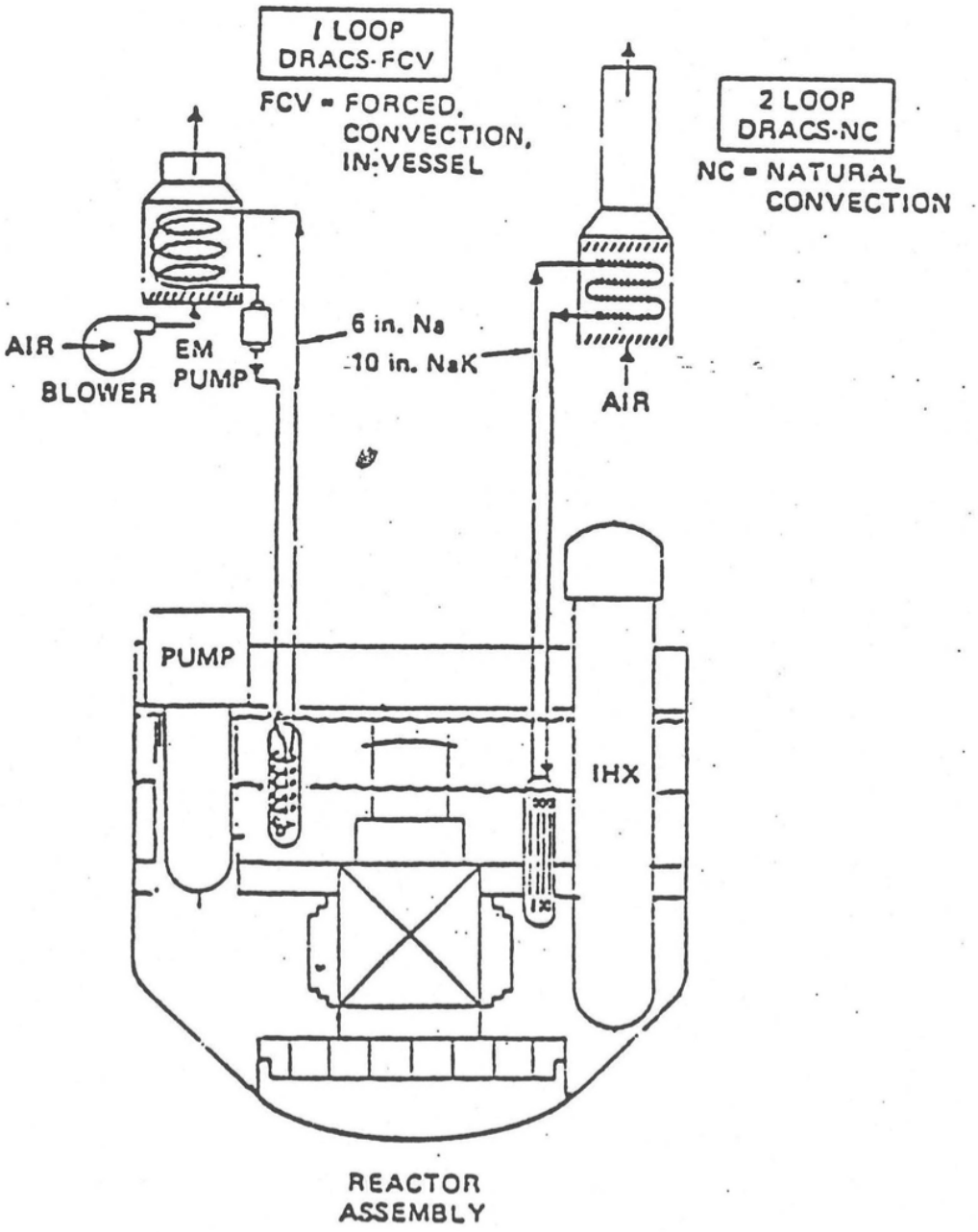


Figure 5.6.1: Shutdown Heat Removal System for a Large Pool Plant.

w = air mass flowrate,

A = area,

ρ_{c} = inlet air density,

ρ_{h} = outlet air density,

K = loss coefficient,

l = height of stack,

SI = stack inlet subscript,

SO = stack outlet subscript,

R = riser subscript

g = acceleration of gravity.

The overall heat transfer coefficient evaluated at the tube outside surface (air side) is

$$U = \left[\frac{r_{\mathrm{o}}}{r_{\mathrm{i}} h_{\mathrm{i}}} + \frac{r_{\mathrm{o}}}{k} \ln \frac{r_{\mathrm{o}}}{r_{\mathrm{i}}} + \frac{1}{h_{\mathrm{o}}} \right]^{-1} , \tag{5.6-3}$$

where

h_{i} = tube inside heat transfer coefficient,

h_{o} = tube outside heat transfer coefficient,

r_{i} = tube inner radius,

r_{o} = tube outer radius,

k = thermal conductivity.

In this implementation of the model, the heat transfer coefficients h_{i} and h_{o} are treated as constants and are input by the user. Strictly, however, their values change with fluid temperature and flowrate. Typically, h_{o} dominates Eq. (5.6-3) and because the air side is a complex geometry of pins and tubes, its value is not accurately known. Therefore, assuming a constant value for h_{o} that is credible should not introduce appreciably more error than the uncertainty that already exists because of its complex geometry.

The basic equations for temperatures are

$$W_{\mathrm{Na}} C_{\mathrm{Na}} \frac{dT_{\mathrm{Na}}}{\mathrm{dz}} + \frac{\mathrm{UA}}{L} \left(T_{\mathrm{a}} - T_{\mathrm{Na}} \right) = 0 \tag{5.6-4}$$

and

$$-W_{\mathrm{a}} C_{\mathrm{a}} \frac{\mathrm{dT_a}}{\mathrm{dz}} + \frac{\mathrm{UA}}{L} \left(T_{\mathrm{Na}} - T_{\mathrm{a}} \right) = 0 \tag{5.6-5}$$

where

T_{Na} = sodium temperature

T_{a} = air temperature

z = elevation

W_{Na} = sodium flow rate

W_{a} = air flow rate

C_{Na} = sodium heat capacity

C_a = air heat capacity

U = heat transfer coefficient

A = total heat transfer area, and

L = height of the heat transfer section

The solution of Eq. (5.6-4) and Eq. (5.6-5) of the form

$$T_{Na} = T_a = ae^{\alpha z} \tag{5.6-6}$$

where a is a constant, and

$$\alpha = \frac{(W_a C_a + W_{Na} C_{Na})\,UA/L}{W_a C_a W_{Na} C_{Na}} \tag{5.6-7}$$

T_{Nab} = sodium outlet temperature, and

T_{ab} = air inlet temperature

The sodium temperature then has the form

$$T_{Na} = b + ce^{\alpha z} \tag{5.6-8}$$

The gravity head is

$$\Delta p_{gr} = \int_0^L \rho_{Na}(z)dz \tag{5.6-9}$$

where ρ_{Na} is the sodium density and $z = 0$ at the bottom of the heat transfer section. The thermal center, z_{tc}, is defined by

$$\Delta p_{gr} = \rho_{Nat}(L - z_{tc}) + \rho_{Nab} z_{tc} \tag{5.6-10}$$

where

ρ_{Nat} = sodium density at the top and

ρ_{Nab} = sodium density at the bottom

If the density varies linearly with temperature, then an equivalent equation for the thermal center is

$$T_{Nat}(L - z_{tc}) + T_{Nab} Z_{tc} = \int_0^L T_{Na}(z)dz \tag{5.6-11}$$

where T_{Nat} is the sodium temperature at the top of the heat transfer section. Carrying out the integration of Eq. (5.6-8) in Eq. (5.6-11) gives

$$z_{tc} = L\left[\frac{1}{(1 - e^{-\alpha L})} - \frac{1}{\alpha L}\right] \tag{5.6-12}$$

For the steady-state initialization, conservation of energy gives

$$W_{Na} C_{Na}(T_{Nat} - T_{Nab}) = W_a C_a (T_{at} - T_{ab}) \tag{5.6-13}$$

Also, Eq. (5.6-6) can give

$$\frac{T_{Nat} - T_{at}}{T_{Nab} - T_{ab}} = x = e \tag{5.6-14}$$

combining Eq. (5.6-13) and Eq. (5.6-14) gives

$$T_{\text{Nab}} = \frac{xW_{\text{a}}C_{\text{a}}}{xW_{\text{a}}C_{\text{a}} - W_{\text{Na}}C_{\text{Na}}} \left[T_{\text{ab}} \left(1 - \frac{1}{x} \right) + \frac{T_{\text{Nat}}}{x} \left(1 - \frac{W_{\text{Na}}C_{\text{Na}}}{W_{\text{a}}C_{\text{a}}} \right) \right] \tag{5.6-15}$$

The above equations are solved by subroutine NAKAIR. SAS4A/SASSYS-1 couples to this routine as follows: Values for liquid metal flowrate at the heat exchanger inlet, the temperature at this point and the inlet air temperature are passed as computed by SAS4A/SASSYS-1 to subroutine NAKAIR. This routine solves the above equations for the temperature drop on the liquid metal side of the heat exchanger and returns this value to SAS4A/SASSYS-1 which uses it to advance to the next time step.

The subroutine is also applicable to the case where the air flowrate is forced. In this case, the known air mass flowrate is used in place of Eq. (5.6-2).

5.6.2 Steady-State Solution

A capability has been added to the SAS4A/SASSYS-1 air dump heat exchanger (DHX) model to provide a steady-state initialization procedure that will calculate the air flow rate required to remove the steady-state power delivered to the heat exchanger. What is done with the steady-state air flow rate in the DHX depends on whether the forced air flow option or the natural circulation air flow option is used. For forced flow, the user-supplied table of air-flow rate vs. time is renormalized by the code to give the correct steady-state value. For natural circulation air flow, the code will calculate and print the inlet orifice coefficient required to give the calculated steady-state air flow rate. The code does not re-set the inlet orifice coefficient to this calculated value, since this is not always what the user would want.

A special case occurs when the user sets the steady-state sodium flow rate through the DHX to zero, indicating that the steady-state heat removal is negligible. In this case, the steady-state air flow rate is set to zero and the thermal center is set to the geometrical center.

For the steady-state initialization of the DHX, the sodium and air inlet temperatures, T_{Nat} and T_{ab}, are known. Also, the sodium flow rate, W_{Na}, is known. Furthermore, the correct value for the sodium outlet temperature, T_{Nab}, is known from initializing the rest of the system. Therefore, the task of the DHX initialization routine is to find the air flow rate which gives the correct value of T_{Nab} in Eq. (5.6-15) This is carried out by iterating on air flow until the correct sodium outlet temperature is found.

5.7 Cover Gas Flow and Pressure Calculations

5.7.1 Introduction

After the hydraulics calculations for the liquid flows and pressures have been completed for a time step, then the liquid temperatures are calculated, and, after they have been completed, the cover gas flows and pressures are computed. The cover gas treatment is a fully implicit calculation, ignoring the inertial effects of the gas, and modeling the flow as quasi-static and isothermal. It may be thought of as a modification of the liquid flow treatment in which the left side of Eq. (5.2-1) is set to zero and θ_1 and θ_2 set to zero and one, respectively, in Eq. (5.2-18).

The gas process may be thought of in the following way. Only compressible volumes containing cover gas are considered. Some may also contain liquid and some only gas. For those containing liquid, the liquid level has risen or dropped, as determined by the hydraulics calculations earlier in the time step, and the cover gas has been compressed or expanded by this action. The compression or expansion is assumed to be adiabatic, and new equilibrium conditions are calculated. Next, heat transfer between the gas and the liquid is computed, assuming that the gas temperature approaches that of the liquid without affecting the liquid temperature. Finally, taking

all compressible volumes with cover gas, the gas flow between compressible volumes through connecting pipes is computed, and the temperatures, pressures, and masses of the gases in the compressible volumes are adjusted appropriately.

5.7.2 Basic Equations

In the compressible volumes with cover gas and liquid, the hydraulics routines provide the rise or fall in the liquid level during the time step, thereby giving the change in the cover gas volume during the time step. Let V_{03} be the volume of the cover gas before the liquid level change and V_3 after the level change. The process is assumed to be adiabatic, and the pressure p'_{03} after the change in terms of the pressure p_{03} before is given by

$$p'_{03} = p_{03}\left(\frac{V_{03}}{V_3}\right)^{\gamma} \tag{5.7-1}$$

and the accompanying new gas temperature T_{g} is given by the ideal gas law as:

$$T_{\mathrm{g}} = p'_{03}\frac{V_3}{m_3 R} \tag{5.7-2}$$

where

γ = the ratio of the specific heat at constant pressure to that at constant volume for the gas

m_3 = the cover gas mass at the beginning of the time step

R = the universal gas constant

Heat transfer between the cover gas and the liquid is also accounted for. Since the heat capacity of the gas is much less than that of the liquid, it is assumed that the gas temperature approaches the liquid temperature with a time constant τ without affecting the liquid temperature. The time constant τ is a user-supplied input quantity. The adjusted gas temperature T_3 is taken as

$$T_3 = T_{\mathrm{g}}\frac{\tau}{\tau + \Delta t} + T_{\mathrm{l}}\frac{\Delta t}{\tau + \Delta t} \tag{5.7-3}$$

where

T_{l} = the temperature of the liquid

Δt = the time step

The new gas temperature means a new gas pressure p_3, which is computed again by the ideal gas law as

$$p_3 = m_3\frac{RT_3}{V_3} \tag{5.7-4}$$

With p_3, V_3, T_3, and m_3 now known for compressible volumes with or without liquid present, we turn to the gas flow through the pipes between compressible volumes. The difference in pressure between two connected compressible volumes causes gas flow from one to the other, and the amount of gas delivered to or withdrawn from a compressible volume in turn modifies the gas pressure in it. As a result, the pressure changes in all of the compressible volumes are solved for simultaneously.

The gas mass flow rate, F_{ij}, from compressible volume i to compressible volume j is approximated at the end of the time step as

$$F_{\mathrm{ij}} = F_{\mathrm{oij}} + F_{1\mathrm{ij}}\Delta p\,(j) + F_{2\mathrm{ij}}\Delta p\,(i) \tag{5.7-5}$$

Here F_{oij} is the gas mass flow rate at the beginning of the time step from compressible volume i to compressible volume j. This flow rate is modified by the pressure changes in the compressible volumes i and j as a result of the flow during the time step. The coefficients of the pressure changes are taken as

$$F_{1\text{ij}} = \frac{-F_{\text{oij}}}{[p(i) - p(j)]} \tag{5.7-6}$$

$$F_{2\text{ij}} = \frac{F_{\text{oij}}}{[p(i) - p(j)]} \tag{5.7-7}$$

where $p(i) - p(j)$ is the pressure difference between compressible volumes i and j at the beginning of the time step. Also

$$\Delta p(i) = p_4(i) - p_3(i) \tag{5.7-8}$$

where

$p_3(i)$ = the gas pressure in compressible volume i at the beginning of the time step

$p_4(i)$ = the gas pressure in compressible volume i at the end of the time step

The flow equation for isothermal unchoked flow of an ideal gas is given by Shapiro [5-3] as:

$$f\frac{L}{D_{\text{h}}} = \frac{1 - \left[\frac{p(i)}{p(j)}\right]^2}{\gamma M^2} \ln\left[\frac{p(i)}{p(j)}\right]^2 \tag{5.7-9}$$

where

f = the Moody friction factor

L = the length of the pipe

D_{h} = the hydraulic diameter of the pipe

$p(i)$ = the inlet pressure

$p(j)$ = the outlet pressure

γ = the ratio of the specific heat at constant pressure to that at constant volume for the gas

M = the Mach number

The Moody [5-2] friction factor is given by:

$$f = 0.0055\left[1 + \left(20000\frac{\varepsilon}{D} + \frac{10^6}{\text{Re}}\right)^{1/3}\right] \tag{5.7-10}$$

where

ε = the pipe roughness

D = the pipe diameter

Re = the Reynolds number

In addition the Reynolds number can be written as

$$\text{Re} = \frac{F_{\text{oij}}}{A}\frac{D}{\mu} \tag{5.7-11}$$

where

A = the pipe area

μ = the viscosity of the gas

and the Mach number is related to the gas mass flow rate F_{oij} by

$$\gamma M^2 = \left(\frac{F_{\text{oij}}}{A}\right)^2 \frac{RT}{\gamma} \tag{5.7-12}$$

For turbulent flow, the gas mass flow rate is determined by an iterative process. With an initial built-in guessed value for f, Eq. (5.7-9) and Eq. (5.7-12) determine a value for F_{oij}, which is used in Eq. (5.7-10) to calculate a new value for f, and the iteration is continued until consistency is achieved. For laminar flow, F_{oij} is determined from Eq. (5.7-9) and Eq. (5.7-12) using the laminar flow value for f of

$$f = \frac{64}{\text{Re}} \tag{5.7-13}$$

5.7.3 Solution

The results of the gas flow between compressible volumes, where a compressible volume may be connected by gas segments to several other compressible volumes, can be written as:

$$p_4(i) = p_3(i)(1 + \varepsilon_{\text{i}}) \tag{5.7-14}$$

$$m_4(i) = m_3(i) + \delta m_{\text{i}} \tag{5.7-15}$$

$$T_4(i) = \frac{p_4(i) V_4(i)}{m_4(i) R} \tag{5.7-16}$$

where

$$\varepsilon_{\text{i}} = \frac{\gamma}{m_3(i) T_3(i)} \sum_{\text{j}} T_{\text{ji}} \Delta m_{\text{ji}} \tag{5.7-17}$$

$$\delta m_{\text{i}} = m_4(i) - m_3(i) = \sum_{\text{j}} \Delta m_{\text{ji}} \tag{5.7-18}$$

$$\Delta m_{\text{i}} = \Delta t \left[F_{\text{oji}} + F_{1\text{ji}} \Delta p(i) + F_{2\text{ji}} \Delta p(j)\right] \tag{5.7-19}$$

and T_{ji} is the temperature of the gas flowing from compressible volume j to compressible volume i. Combining Eq. (5.7-14), Eq. (5.7-8), Eq. (5.7-17), and Eq. (5.7-19) yields a matrix equation of the form

$$\begin{pmatrix} c_{11} & c_{12} & \ldots\ldots \\ c_{21} & c_{22} & \ldots\ldots \\ \ldots\ldots & \ldots\ldots & \ldots\ldots \\ \ldots\ldots & \ldots\ldots & \ldots\ldots \end{pmatrix} \begin{pmatrix} \Delta p(1) \\ \Delta p(2) \\ \ldots\ldots \\ \ldots\ldots \end{pmatrix} = \begin{pmatrix} d_1 \\ d_2 \\ \ldots\ldots \\ \ldots\ldots \end{pmatrix} \tag{5.7-20}$$

where

$$c_{\text{ij}} = \delta_{\text{ij}} \left[1 - \frac{p_3(i) \gamma \Delta t}{m_3(i) T_3(i)} \sum_{\text{k}} T_{\text{ki}} F_{1\text{ki}}\right] - \frac{p_3(i) \gamma \Delta t}{m_3(i) T_3(i)} T_{\text{ji}} F_{2\text{ji}} \tag{5.7-21}$$

and

$$d_{\text{i}} = \frac{p_3(i) \gamma \Delta t}{m_3(i) T_3(i)} \sum_{\text{j}} T_{\text{ji}} F_{\text{oji}} \tag{5.7-22}$$

Here δ_{ij} is the Kronecker delta.

The matrix equation 5.7-20 is solved by Gaussian elimination for the $\Delta p\,(i)$. These are added to the pressures at the beginning of the time step to obtain the corresponding pressures at the end of the time step. The cover gas mass at the end of the time step $m_4\,(i)$ in each compressible volume is computed using Eq. (5.7-18) and Eq. (5.4-31), and finally the gas temperatures at the end of the time step $T_4\,(i)$ are computed for each compressible volume using the equation

$$T_4\,(i) = \frac{p_4\,(i)\,V_4\,(i)}{m_4\,(i)\,R} \tag{5.7-23}$$

All of the final values are then stored in COMMON blocks.

A solution algorithm for the entire cover gas treatment is given in Section 5.16.3.

5.8 Overall Solution and Time Step Control

5.8.1 Order of Calculation in PRIMAR

The computational sequence used in a time step in the primary and intermediate loops is as follows:

1. Adjust the inlet and outlet plenum pressures for any errors between the estimated and the calculated core flows in the last step.
2. Calculate b_o, b_1, and b_2 for the compressible volumes in the primary loop.
3. Calculate the contributions to a_o, a_1, a_2, and a_3 for each element in a liquid segment and sum them for all of the liquid segments in the primary loop.
4. Calculate contributions to c_{ij} and d_j from all of the segments in the primary loop.
5. Add the contributions to c_{ij} and d_j from the estimated core flow.
6. Solve for Δp.
7. Calculate Δw.
8. Repeat steps 2, 3, 4, 6 and 7 for the intermediate loops, if any are present.
9. Repeat steps 2, 3, 4, 6 and 7 for the DRACS loops, if any are present.
10. Calculate the liquid temperatures.
11. Recalculate the compressible volume pressures with the new liquid temperatures.
12. Calculate the cover gas flows and the cover gas pressures.

In this sequence of events, the liquid flow hydraulics calculations are done first, followed by the liquid temperature calculations and then by the gas flow and temperature calculations. In order to reduce the sizes of the matrix equations that must be solved, the hydraulics calculations are done for the primary loop first, then for the intermediate loops, and finally for the DRACS loops, if any are present. Adiabatic compression of the cover gases is accounted for during the initial hydraulics calculations, but heat transfer to the gas and gas flows through connecting pipes are not accounted for until the gas calculations at the end of the computational sequence for a time step.

5.8.2 PRIMAR Time Step Size

The initial PRIMAR time-step size is entered as an input quantity. Then the time-step size is determined by the coupling with the coolant dynamics calculations for the core subassemblies. PRIMAR is called to calculate a new time step before the core coolant dynamics routines are called and, as a result, part of the PRIMAR calculation consists in estimating the new core flows based on information supplied by the coolant dynamics routines. Before the start of voiding, the core flows can be estimated accurately in PRIMAR because the single-phase subassembly flow calculations are relatively simple. Consequently, the time step can be fairly large before the start of voiding. After the start of voiding, the core flows are strongly influenced by the rapidly varying pressures in the voiding region, and the PRIMAR time-step size must be cut back to 20 milliseconds or less.

If PRIMAR predicts a large change in the inlet plenum pressure during a time step, the step size must be reduced and the quantities determined in the time step recalculated. Alternatively, if the pressure, temperature, or flow in the primary or intermediate loop change rapidly, but do not have a large effect on the inlet or outlet plenum pressures and temperatures, then the PRIMAR time step is subdivided into smaller intervals for the PRIMAR calculations without requiring additional coolant dynamics calculations in the core subassemblies.

5.9 Steady-State Initialization

The steady-state initialization for PRIMAR-4 is based on finding a steady-state thermal hydraulic solution that is consistent with the user-supplied input for parameters such as reactor power, flow, coolant inlet temperature, and coolant outlet plenum pressure. The core channels are initialized first, leading to the temperatures and pressures in the inlet and outlet coolant plenums. Then the initialization works outward through the primary heat-transport loops, the *IHX's*, the intermediate heat-transport loops, and finally, the steam generators.

The primary-loop temperature initialization consists mainly of setting the temperatures of all elements and compressible volumes in the hot side of the loop to the coolant outlet plenum temperature, and setting the components in the cold side of the loop to the inlet plenum temperature. Intermediate heat exchangers and bypass channels are separate cases, since the temperatures change from one end to the other end of these elements. If necessary, the liquid segment flows are adjusted so that the liquid flow into each compressible volume balances the flow out. After the primary-loop temperatures are set, the pressure drops through all elements are calculated, and the pump heads are set to the values necessary to achieve a pressure balance. If a parallel flow path is detected, an adjustment to an element loss coefficient may be made. These adjustments will be reported in the standard output file. The magnitude of these adjustments can be limited using **ADJKTOL**, and the adjustments cannot result in a negative loss coefficient. If **ADJKTOL** is set to a positive value and SAS determines an adjustment to **G2PRDR** that is larger than the defined tolerance is required, SAS will report an error and the simulation will be terminated. If SAS determines an adjustment to **G2PRDR** that results in a negative loss coefficient is required, SAS will report a warning and not make the adjustment.

After the main primary-loop initialization calculations, the *IHX* primary and intermediate side flow rates are known, and the primary side inlet and outlet temperatures are known. Then, an iteration procedure is used to find intermediate side inlet and outlet temperatures that provide a consistent steady-state solution. The detailed node-by-node temperature distributions within the *IHX* are also calculated in this process.

After the *IHX* temperature distributions have been calculated, the temperatures in the intermediate loop between the *IHX* outlet and the steam generator inlet are set to the *IHX* outlet temperature, and the temperatures between the steam generator outlet and the *IHX* inlet are set to the *IHX* inlet temperature. Pressure drops are calculated for each element; and pressures at each point are calculated, starting from a user-specified *IHX* inlet pressure. As in the primary loop, the intermediate loop pump head is set to the value necessary to achieve a pressure balance.

After the intermediate loop initialization, the steam generators are initialized, using the known sodium side inlet and outlet temperatures and flow rate.

The steady-state gas flow rates are all set to zero, and cover-gas pressures are initially set to user-supplied values.

5.9.1 Core Channels, Inlet Plenum, and Outlet Plenum

Since the core channel calculations are described in detail in Chapter 3, they are only summarized here. The user supplies the steady-state values for the coolant outlet plenum pressures, the coolant inlet temperatures, the coolant flow for each channel, and the power for each channel, as well as geometric and hydraulic information. The code then calculates the coolant outlet temperature and an inlet pressure for each channel, as well as temperature and pressure distributions within the channels. In general, each channel will have a different inlet pressure, so for each inlet plenum the channel with the highest inlet pressure is picked as the reference, and the orifice coefficients at the bottom of the first reflector zone in each other channel are adjusted to give a common inlet pressure. This common inlet pressure determines the coolant inlet plenum pressure.

For each outlet plenum, the average steady-state temperature is a mixed mean of the outlet temperatures of all the core channels and all the bypass channels emptying into the plenum. The core model provides steady-state channel outlet temperatures, but the loop model is designed to begin with known temperatures in the inlet and outlet plena of the core channels and from these calculate temperatures throughout the remainder of the plant, and so outlet temperatures for the bypass channels are not available at the point at which the outlet plenum temperatures need to be calculated. The code works around this as follows. To a particular outlet plenum ICV are attached N channels and M bypass channels. The mixed-mean outlet temperature, T_{mm}, of the core channels is given by

$$T_{\text{mm}}\left(\text{ICV}\right) = \frac{\sum_{\text{ic}=1}^{N} W_{\text{J(ic)}} T_{\text{exp}}\left(J\left(ic\right)\right)}{\sum_{\text{ic}=1}^{N} W_{\text{J(ic)}}} \tag{5.9-1}$$

where $T_{\text{exp}}\left(J\left(ic\right)\right)$ is the steady-state outlet temperature for core channel $J\left(ic\right)$ and $w_{\text{J}}\left(ic\right)$ is the steady-state flow through channel $J\left(ic\right)$. The model assumes that the difference between $T_{\text{mm}}\left(\text{ICV}\right)$ and the average inlet temperature of the N core channels, $T_{\text{in}}\left(\text{ICV}\right)$, is proportional to the ratio of power to flow in the channels, or

$$T_{\text{mm}}\left(\text{ICV}\right) - T_{\text{in}}\left(\text{ICV}\right) = K\frac{f_{\text{pr}}\left(\text{ICV}\right)}{f_{\text{fl}}\left(\text{ICV}\right)}\ , \tag{5.9-2}$$

where K is the constant of proportionality, $f_{\text{pr}}\left(\text{ICV}\right)$ is the ratio of the power in the core channels attached to outlet plenum ICV to the power in all channels (core + bypass) attached to plenum ICV, and $f_{\text{fl}}\left(\text{ICV}\right)$ is the ratio of the flow in the core channels attached to outlet plenum ICV to the flow in all channels (core plus bypass) attached to plenum ICV. The model also assumes that the difference between the average temperature of plenum ICV, $T_{\text{out}}\left(\text{ICV}\right)$, and the average inlet temperature of the N core channels and the M bypass channels, $T_{\text{in,total}}\left(\text{ICV}\right)$, is proportional to the ratio of power to flow and that the constant of proportionally is the same as for the core channels alone, so that

$$T_{\text{out}}\left(\text{ICV}\right) - T_{\text{in,total}}\left(\text{ICV}\right) = K\ . \tag{5.9-3}$$

The average inlet temperatures are calculated as

$$T_{\text{in}}\left(\text{ICV}\right) = \frac{\sum_{\text{ic}=1}^{N} W_{\text{J(ic)}} C_{\text{pJ(ic)}} T_{\text{in,channel}}\left(J\left(ic\right)\right)}{\sum_{\text{ic}=1}^{N} W_{\text{J(ic)}} C_{\text{pJ(ic)}}} \tag{5.9-4}$$

and

$$T_{\text{in,total}}\left(\text{ICV}\right) = \frac{\sum_{\text{ic}=1}^{N} W_{\text{J(ic)}} C_{\text{pJ(ic)}} T_{\text{in,channel}}\left(J\left(ic\right)\right) + \sum_{\text{jc}=1}^{M} W_{\text{K(jc)}} C_{\text{pK(jc)}} T_{\text{in,bypass}}\left(K\left(jc\right)\right)}{\sum_{\text{ic}=1}^{N} W_{\text{J(ic)}} C_{\text{pJ(ic)}} + \sum_{\text{jc}=1}^{M} W_{\text{K(jc)}} C_{\text{pK(jc)}}}\ , \tag{5.9-5}$$

where C_{p} is the specific heat of the fluid at the entrance to a channel. The inlet temperatures and the flows are input by the user, and the specific heats are computed from the equation of state. The fractions f_{pr} and f_{fl} are calculated from user-input power and flow information, and T_{mm} is computed from Eq. (5.9-1), so the only

unknown quantities in Eq. (5.9-2) and Eq. (5.9-3) are K and T_{out}. Eliminating K between the two equations gives

$$T_{\text{out}}\left(\text{ICV}\right) - T_{\text{in,total}}\left(\text{ICV}\right) = \frac{T_{\text{mm}}\left(\text{ICV}\right) - T_{\text{in}}\left(\text{ICV}\right)}{\frac{f_{\text{pr}}(\text{ICV})}{f_{\text{fl}}(\text{ICV})}} \,, \tag{5.9-6}$$

or

$$T_{\text{out}}\left(\text{ICV}\right) = T_{\text{mm}}\left(\text{ICV}\right)\frac{f_{\text{fl}}\left(\text{ICV}\right)}{f_{\text{pr}}\left(\text{ICV}\right)} + T_{\text{in,total}}\left(\text{ICV}\right) - T_{\text{in}}\left(\text{ICV}\right)\frac{f_{\text{fl}}\left(\text{ICV}\right)}{f_{\text{pr}}\left(\text{ICV}\right)} \,. \tag{5.9-7}$$

5.9.2 Primary Heat Transport System

After the core channel initialization, and after the temperatures and pressures have been set for the inlet and outlet plenums, the rest of the primary heat-transport loop is initialized. This involves setting temperatures for all compressible volumes and liquid elements, liquid pressures and liquid-cover gas interface heights for compressible volumes, and flow rates for all liquid segments. Flow rates for gas segments are initialized to zero, and the user specifies the initial cover-gas pressure for each compressible volume.

A two-pass procedure is used for initializing the primary heat-transport system other than core channels, the inlet plenums, and the outlet plenums. In the first pass, the temperatures are set for all compressible volumes and elements, and the gravity head and pressure drop for each element other than a pump is set. Also the liquid segment flows are adjusted, if necessary, to make the flow into each compressible volume equal the flow out. In the second pass, the compressible volume pressures and pump heads are set.

For each pass through the primary heat-transport system, a table is constructed containing the compressible volume numbers of the compressible volumes whose temperatures or liquid pressures are already known. The inlet and outlet plenums are the first entries in this table. Compressible volumes are treated one at a time in the order in which they occur in the table, and new entries are added to the table as temperatures or pressures of new compressible volumes are found.

For the temperatures pass, the first step in the treatment of a compressible volume is to adjust the flows in the liquid segments attached to the compressible volume so that the flow in matches the flow out. The user supplies the steady-state flow rates for each liquid segment, but if the user-supplied values are not consistent then the code adjusts some of them to obtain consistency. If the pressure drops in a liquid segment have already been set, then a flag is set for that segment; and its flow can no longer be changed, since changing flows would require re-calculating the pressure drops. Therefore, any flow adjustment is made for liquid segments whose flows have not been fixed. A fixed net flow, w_{fixed}, is calculated by summing the flows of all fixed segments attached to the compressible volume, with a positive sign if the flow is into the compressible volume, or a negative sign if the flow is out of the compressible volume. Similarly, an adjustable net flow, w_{adj}, is calculated by summing over the remainder of the liquid segments attached to the compressible volume. Then a factor, f_{w}, is calculated as

$$f_{\text{w}} = -\frac{w_{\text{fixed}}}{w_{\text{adj}}} \tag{5.9-8}$$

and the flow for each unfixed liquid segment attached to the compressible volume is multiplied by f_{w}.

The next step in the treatment of a compressible volume consists of following each liquid segment attached to the compressible volume and setting the temperatures for all of the elements until either the other end of the liquid segment is reached or a stop is made at an IHX or at the element representing the core channels. Initially, the compressible volume temperature is used for setting the temperatures of the elements in a segment; but if a bypass channel is encountered, then the bypass channel initialization routine is called, and the bypass outlet temperature is used for setting the temperatures of any additional elements in the segment. The steady-state gravity head and pressure drop for an element are set as soon as the temperature is set. The liquid segment containing the core channels is ignored, since the core channels have already been initialized. The initialization

procedure used for an IHX requires knowing both the primary side inlet temperature and the primary side outlet temperature; so if an IHX is encountered before the temperatures of the compressible volumes at both ends of the liquid segment are known, then the remainder of this segment is skipped until later, when it will be approached from the other end.

If a liquid segment is followed to the end without encountering the core channels or an IHX, then the compressible volume at the other end of the segment is checked. If its temperature has not already been set, then it is now set to the current temperature, and the new compressible volume is added to the compressible volume table. In this manner, all of the elements, liquid segments, and compressible volumes in the primary loop will eventually be initialized if they are all inter-connected, and if two IHX's are not connected in series. If two or more IHX's are connected in series, then the user must specify the steady-state temperature drops across all but one of them. Similarly, if two or more pumps are connected in series, then the user must specify the steady-state pump heads for all but one of them.

For the second pass through the primary heat-transport system, the compressible volume table is started over, with the inlet and outlet plenums again providing the first entries in the table. Now the table contains only compressible volumes whose pressures are known. In this pass, each liquid segment attached to a compressible volume is again treated. The pressure drops and gravity heads for all of the elements, other than pumps, in a liquid segment are summed. If the liquid segment contains a pump, and if the pressure has already been determined for the compressible volumes at both ends of the liquid segment, then the pump head is set to the value needed to balance the pressures. If the segment contains a pump, and if the pressure in one of the compressible volumes at the end of the liquid segment has not been set yet, then this liquid segment is skipped until later, when the compressible volume pressure has been set. If there are no pumps in the liquid segment, and if the pressure in the compressible volume at one end of the liquid segments has not yet been set, then the pressure in this compressible volume is set, based on the pressure in the compressible volume at the other end of the liquid segment and the sum of the pressure drops and gravity heads through the segment. This compressible volume is then added to the table. If there is no pump in the liquid segment, and if the pressures in the compressible volumes have already been set, then the orifice-type pressure-drop coefficient for the first element in the liquid segment is adjusted to give the proper pressure balance.

During the first pass, after the primary side inlet and outlet temperatures and the primary side flow rate have been determined, an IHX initialization routine is called to initialize the primary and intermediate side IHX temperatures and gravity heads. This routine is described in Section 5.9.4.3.

5.9.3 Intermediate Heat Transport System

The steady-state initialization of the intermediate heat-transport system is similar to that of the primary heat-transport system, except that the intermediate loop treatment starts from the IHX and works out. The IHX intermediate side inlet and outlet temperatures have been determined by the IHX initialization routines. Also, since the IHX flow rates were used in the IHX routines, the flow rate through the segment containing the intermediate side of the IHX is fixed at the user-supplied value for this segment. The pressure calculations start from a user-supplied IHX inlet pressure.

Three passes through the intermediate loop compressible volumes are made. In the first pass, the liquid segment flows are adjusted so that the flow into each compressible volume equals the flow out. This is exactly the same method used for the primary loop, with sums over fixed and adjustable flows and modifications of the adjustable flows. In the second pass, temperatures, gravity heads, and pressure drops are set for each element except for pumps. In the third pass, pump heads and compressible volume pressures are set. The second and third passes use the same type of compressible volume tables used for the primary loop. These passes use the same kind of calculations as the corresponding primary loop passes, except that the calculations start in the middle of a segment at the IHX, and the first compressible volumes added to the table are those at the ends of the IHX segment.

One difference between the primary and intermediate loop initialization procedures is that, in the intermediate

loop, it is necessary to consider the possibility of two steam generators connected in series, which is a situation that occurs when an evaporator-superheater combination is used. For such a case, the user specifies the fraction of the total sodium-side steady-state temperature drop that occurs across the evaporator.

5.9.4 Individual Components

For each element, other than the core channels, of each liquid segment in the primary and intermediate heat-transport systems, a steady-state initialization routine is called. These routines compute pressure drops, except for pumps, and gravity heads. Temperature distributions within the component are also calculated for IHXs and bypass channels. Also, routines are called to initialize the temperatures, liquid pressures, gas pressures, and liquid-gas interface elevations of compressible volumes. These routines for individual components are described below.

5.9.4.1 Pressure Drop, Subroutine PRESDR

For all elements other than core channels and pumps, subroutine PRESDR is called to compute the pressure drop due to friction, bends, orifices, and density change. The pressure drop for element *k* in segment *i* is calculated as

$$\begin{aligned}\Delta p_{\mathrm{e1}}(k) = w(i)\left\{\frac{f\,|w(i)|}{2\overline{\rho}A_{\mathrm{K}}^2}\left[\frac{L_{\mathrm{k}}}{D_{\mathrm{h}}(k)} + N_{\mathrm{b}}(k)\left(\frac{L}{D}\right)_{\mathrm{b}}\right]\right. \\ \left. + \frac{w(i)}{A_k^2}\left[\frac{1}{\rho_{\mathrm{out}}(k)} - \frac{1}{\rho_{\mathrm{in}}(k)}\right] + \frac{|\,w(i)|}{2\overline{\rho}A_{\mathrm{k}}^2}G_{\mathrm{2pr}}(k)\right\}\end{aligned} \tag{5.9-9}$$

The first term is the friction, as enhanced by bends. N_{b} is the number of bends in the element, and $(L/D)_{\mathrm{b}}$ is the effective length/hydraulic diameter per bend. The second term is an acceleration term due to density changes. In this term, ρ_{in} and ρ_{out} are the sodium densities at the inlet and outlet of the element. The last term represents orifice-type pressure drops proportional to the square of the flow rate, and G_{2pr} is the dimensionless loss coefficient for this term. The average density is the average of ρ_{in} and ρ_{out}.

The Moody correlation [5-2] is used for the friction factor for Reynolds numbers, Re, above 1082, whereas a laminar friction factor is used for lower Reynolds numbers:

$$f = \begin{cases} 0.0055\left[1 + \left(20000\frac{\varepsilon(k)}{D_{\mathrm{n}}(k)} + \frac{10^6}{\mathrm{Re}}\right)^{\frac{1}{3}}\right] & \text{if Re} \geq 1082 \\ \frac{64}{\mathrm{Re}} & \text{if Re} < 1082 \end{cases} \tag{5.9-10}$$

where $\varepsilon(k)$ is the surface roughness. The value of 1082 was picked for the transition between laminar and turbulent friction factors because at this value both correlations give the same result if $\varepsilon/D_{\mathrm{h}}$ is small.

5.9.4.2 Pipe

For a pipe, the pressure drop is computed as indicated in Section 5.9.4.1 above. The gravity head is calculated as

$$\Delta p_{\mathrm{gr}}(k) = \rho_{\mathrm{k}}\left[z_0(k) - z_{\mathrm{i}}(k)\right] g \tag{5.9-11}$$

where k is the element number, ρ_{k} is the sodium density, z_{i} and z_0 are the inlet and outlet elevations, and g is the acceleration of gravity.

5.9.4.3 IHX

The steady-state initialization of an IHX depends on whether the detailed IHX model or the simple table loop-up model is used. In either case, the pressure drop is calculated in subroutine PRESDR, as described in Section 5.9.4.1. The temperature calculations for the detailed model are described in Section 5.4.2.5. For the simple table look-up model, the gravity head in both the steady-state and the transient is calculated using Eq. (5.4-104).

5.9.4.4 Bypass Channel

The steady-state temperature solution for the bypass channel is described in Section 5.4.3. The pressure drop is calculated in subroutine PRESDR, as described in Section 5.9.4.1.

5.9.4.5 Steam Generator

The table look-up steam generator model is treated the same as the table look-up IHX model. Subroutine PRESDR, as described in Section 5.9.4.1, calculates the pressure drop, and the gravity head is calculated using Eq. (5.4-104). The detailed steam generator model is described in Chapter 7.

5.9.4.6 Compressible Volumes

If compressible volume i contains liquid, then the steady-state liquid pressure, $p_{\mathrm{l}}(i)$, and temperature, $T_{\mathrm{l}}(i)$, are determined by the procedure described in Section 5.9.2 or Section 5.9.3. Then the initial liquid volume, $V_{\mathrm{l}}(i)$, is

$$V_{\mathrm{l}} = \begin{cases} V_{\mathrm{tot}}(i) & \text{if there is no gas} \\ V_{\mathrm{tot}}(i) - V_{\mathrm{go}}(i) & \text{if there is gas} \end{cases} \tag{5.9-12}$$

where $V_{\mathrm{tot}}(i)$ is the total volume, and $V_{\mathrm{go}}(i)$ is the initial gas volume. The values of $V_{\mathrm{tot}}(i)$ and $V_{\mathrm{go}}(i)$ are specified by the user, whereas $V_{\mathrm{l}}(i)$ is computed by the code. The liquid mass, $m_{\mathrm{l}}(i)$, is then computed as

$$m_{\mathrm{l}}(i) = \rho_{\mathrm{l}}(i) V_{\mathrm{l}}(i) \tag{5.9-13}$$

where ρ_{l} is the liquid density. The wall temperature is set equal to the liquid temperature.

If there is only liquid in the compressible volume, then the reference pressure, $p_{\mathrm{r}}(i)$, is set equal to the steady-state pressure $p_{\mathrm{l}}(i)$; and the reference temperature, $T_{\mathrm{r}}(i)$, is set equal to the steady-state liquid temperature, $T_{\mathrm{l}}(i)$.

The liquid pressure within a compressible volume varies with elevation due to the gravity head. The pressure $p_{\mathrm{l}}(i)$ is defined at the reference elevation $z_{\mathrm{l}}(i)$, so the pressure at any elevation z, within the liquid is given by

$$p(z) = p_{\mathrm{l}}(i) + (z_{\mathrm{l}}(i) - z)\rho_{\mathrm{l}} g \tag{5.9-14}$$

where g is the acceleration of gravity. The steady-state liquid-gas interface elevation, $z_{\mathrm{i}}(i)$, is the elevation at which the user-supplied gas pressure, $p_{\mathrm{g}}(i)$, is equal to the liquid pressure:

$$z_{\mathrm{i}}(i) = z_{\mathrm{l}}(i) + \frac{p_{\mathrm{l}}(i) - p_{\mathrm{g}}(i)}{\rho_{\mathrm{l}}(i) g} \tag{5.9-15}$$

Also, for use in the transient calculations, a minimum elevation, $z_{\mathrm{m}}(i)$, is defined as

$$z_{\mathrm{m}}(i) = z_{\mathrm{i}}(i) - \frac{V_{\mathrm{l}}(i)}{A_{\mathrm{i}}(i)} \tag{5.9-16}$$

where $A_{\mathrm{i}}(i)$ is the liquid-gas interface area. The minimum elevation, $z_{\mathrm{m}}(i)$, is the "bottom" of the compressible volume; if the liquid-gas interface elevation drops to z_{m}, then there will be no liquid left in the compressible volume.

If the compressible volume contains only gas, then the gas volume, $V_{\mathrm{g}}(i)$, is equal to the total volume, $V_{\mathrm{tot}}(i)$. The initial gas and wall temperatures are set to a user-supplied value.

In any compressible volume containing gas, the initial gas mass is calculated as

$$m_{\mathrm{g}}(i) = \frac{p_{\mathrm{g}}(i) V_{\mathrm{g}}(i)}{R_{\mathrm{g}} T_{\mathrm{g}}(i)} \tag{5.9-17}$$

where R_{g} is the gas constant.

5.9.4.7 Pumps

The steady-state pump head, flow, and temperature for each pump are determined by the procedure in Section 5.9.2 or Section 5.9.3. Then, the gravity head is calculated using Eq. (5.9-11). If a centrifugal pump option is used then the pump speed required to obtain the required steady-state pump head with the specified flow is obtained by iteration, as described in Section 5.9.4.2.

An electromagnetic pump is treated as a loss-less pump in series with a pipe. The pump head is calculated as

$$\Delta p_{\mathrm{p}}(t) = H_{\mathrm{so}} F_{\mathrm{p}}(t) \left[1 - \frac{w}{\rho A V_{\mathrm{s}}}\right] - \Delta p_{\mathrm{pipe}} \tag{5.9-18}$$

where H_{so} is the steady-state stall head, $F_{\mathrm{p}}(t)$ is a user-supplied function for the time dependence of the stall head, V_{s} is the synchronous velocity of the magnetic field, and Δp_{pipe} is the pipe-type pressure drop, given by Eq. (5.9-9). For the steady-state solution $F_{\mathrm{p}}(t)$ is 1.0. After the steady-state pump head and flow are known, the steady-state stall head is obtained from Eq. (5.9-12) as

$$H_{\mathrm{so}} = \frac{\Delta p_{\mathrm{p}}(t=0) + \Delta p_{\mathrm{pipe}}}{1 - \frac{w}{\rho A V_{\mathrm{s}}}} \tag{5.9-19}$$

5.9.5 Null Transient for Steady-State Initialization

In order to eliminate a spurious reactivity insertion at the start of SAS4A/SASSYS-1 transient calculations due to the neglect of component-to-component heat transfer, RVACS heat removal, and direct element heating in the PRIMAR-4 steady-state initialization, a new capability to bring the PRIMAR-4 temperatures into equilibrium before the start of the transient calculations has been added to the code. This new capability requires small modifications to the code and uses a relatively small amount of computer time to provide significantly better initial conditions at the start of the transient calculations.

The current PRIMAR-4 steady-state initialization does not account for component-component heat transfer and RVACS heat removal, although these effects are accounted for during the transient calculations. In the current LMR designs, component-component heat transfer and RVACS heat removal have a significant impact on vessel wall temperatures; and changes in vessel wall temperatures can lead to large reactivity changes. Therefore, at the start of a transient calculation the vessel wall temperatures will change as they approach a new state in equilibrium with component-to-component heat transfer and RVACS heat removal; and these temperature changes will add a significant amount of spurious reactivity.

Re-writing the PRIMAR-4 steady-state initialization routines to provide a direct equilibrium temperature solution including arbitrary component-to-component heat transfer would require re-formulating the whole steady-state initialization algorithm and would require completely re-writing many of the routines. Also, a direct equilibrium solution could require a large amount of computer time. Therefore, a simpler approach was taken. The

current PRIMAR-4 steady-state initialization routines are used to provide coolant flow rates and a starting point for the temperatures. Then a null transient is run to set the temperatures before the start of the regular transient. During the null transient the transient heat transfer routines are used, including component-component heat transfer, RVACS heat removal, and direct element heating.

Null Transient

During the null transient, only the PRIMAR-4 transient heat transfer routines are called, and everything except the PRIMAR-4 temperatures (excepting the inlet plenum temperature) is held constant. Because changing temperatures in *equivalent circuit pumps* with temperature-dependent circuit parameters will impact their steady-state initialization, the related pump initialization routines may also called. A user-specified time step size is used for the null transient. The user also specifies the number of steps to be used in the null transient and specifies a convergence criterion. The null transient is run until either the specified number of steps have been run or until the maximum fractional change from step to step in component-to-component heat transfer or RVACS heat removal is less than the specified convergence criterion. After the completion of the null transient, the reactivity components are set to zero before the start of the regular transient calculation.

During the null transient, the heat removal from steam generators and table look-up IHXs is reduced by an amount equal to the RVACS heat removal and increased by an amount equal to the direct element heat generation so that the total heat removal rate matches the total power generation rate.

There are a number of advantages to this null transient capability. It allows the temperatures to converge toward the correct equilibrium values, and it eliminates spurious reactivity insertion during the regular transient due to non-equilibrium initial temperatures for the vessel wall. Also, the null transient does not require much computing time. Each time step is fast because all core channel calculations and PRIMAR-4 flow and pressure calculations are bypassed. The time steps can be large, without any stability limitations due to flow rate calculations, although some heat transfer time step stability limits still apply.

The null transient approach has some limitations. One limitation is that temperature changes during the null transient change gravity heads. The PRIMAR-4 steady-state initialization adjusts pump heads and orifice coefficients so that the pressures and flows are in equilibrium at the start of the null transient. Currently, no additional adjustments are made at the end of the null transient; so at the beginning of the regular transient the liquid pressures and flows might not be exactly in equilibrium. At nominal flow rates, the gravity heads are a small part of the over-all pressure balances; so moderate temperature changes during the null transient will usually have only a very small impact on the pressure balances at the start of the regular transient. A second limitation is that it can take a long time for the null transient to converge to an equilibrium situation if there are long thermal time constants in the system. This limitation can usually be gotten around by using fictitiously low heat capacities for some components during the null transient and then changing the heat capacities to their proper values on a restart before the start of the regular transient.

Time Step Limitations

Most of the temperature calculations in PRIMAR-4 are semi-implicit or fully implicit, and there are no time step stability limits imposed by most of these calculations, but in the component-to-component heat transfer calculations there is at least one place where explicit forward differencing limits the time step size. For component-to-component heat transfer from a source with temperature T_{s} to a compressible volume liquid, the equation solved is of the form

$$M_{\mathrm{l}} C_{\mathrm{l}} \frac{\mathrm{dT_l}}{\mathrm{dt}} = hA\left(T_{\mathrm{s}} - T_{\mathrm{l}}\right) + \text{ other terms} \tag{5.9-20}$$

where M_{l}, C_{l}, and T_{l} are the mass, specific heat, and temperature of the liquid, and hA is the heat transfer coefficient times area for the component-to-component heat transfer. The evaluation of the $hA\left(T_{\mathrm{s}} - T_{\mathrm{l}}\right)$ term is partly explicit, so for stability the maximum time step that can be used is of the order of , where

$$\tau = \frac{M_{\mathrm{l}} C_{\mathrm{l}}}{hA} \tag{5.9-21}$$

For typical cases, this maximum time step is quite large (tens or hundreds or thousands of seconds), but in some cases it can be small enough to be a significant limitation.

5.10 PRIMAR-1 Option

PRIMAR-4 contains both a simple PRIMAR-1 type option and the more detailed PRIMAR-4 treatment. The PRIMAR-1 option supplies only the minimum quantities required to drive the subassembly coolant dynamics module. It is often used when the more detailed treatment is not required. The minimum information consists of the inlet and outlet plenum pressures, the subassembly inlet temperatures, and the outlet reentry temperatures. All of these quantities are supplied as a function of time by PRIMAR-1 from user-supplied information, independent of what is happening in the core or in the rest of the primary loop.

There are two available options for determining the inlet pressure. The first option is based on a normalized pump head and the second option is based on a normalized flow rate through channel **IFLOW**.

5.10.1 Defined Pump Head

When **IFLOW** = 0, the inlet pressure is based on a normalized pump head. The inlet pressure $p_{\text{in}}(t)$ is calculated as

$$p_{\text{in}}(t) = p_{\text{x}} + p_{\text{gr}} + \Delta p_{\text{p}}(t) \tag{5.10-1}$$

where p_{x} is the user-specified exit plenum pressure, **PX**, p_{gr} is the gravity head, and $\Delta p_{\text{p}}(t)$ is the pump head. The gravity head is calculated as

$$p_{\text{gr}} = \left[\rho_{\text{HOT}}\left(z_{\text{PU}} - z_{\text{IHX}}\right) + \rho_{\text{COLD}}\left(z_{\text{IHX}} - z_{\text{PL}}\right)\right] g \tag{5.10-2}$$

where ρ_{HOT} is the average steady-state outlet density, ρ_{COLD} is the average steady-state inlet density, z_{PU} is the reference height at which the outlet plenum pressure is calculated, **ZPLENU**, z_{PL} is the reference height at which the inlet plenum pressure is calculated, **ZPLENL**, and z_{IHX} is the reference height of the thermal center of the intermediate heat exchanger, **ZIHX**.

The pump head is calculated as

$$\Delta p_{\text{p}}(t) = \Delta p_{\text{o}} f_{\text{p}}(t) \tag{5.10-3}$$

where Δp_{o} is the steady-state pump head, taken as

$$\Delta p_{\text{o}} = p_{\text{in}}(t = 0) - p_{\text{gr}} - p_{\text{x}} \tag{5.10-4}$$

and $f_{\text{p}}(t)$ is the user-supplied time dependence of the pump head, which should be normalized so that

$$f_{\text{p}}(t = 0) = 1.0 \tag{5.10-5}$$

There are two options for specifying $f_{\text{p}}(t)$. One option is for the user to supply a table of f_{p} as a function of time. This table is defined in **PRETAB** and **PRETME** or a *function block*, depending on the value of **NPRES**. With this option, the code interpolates linearly between table entries.

The other option is for the user to supply the three coefficients p_{d}, p_{d1}, p_{d2} for use in the equation

$$f_{\text{p}}(t) = \exp\left[-\left(p_{\text{d}} t + p_{\text{d1}} t^2 + p_{\text{d2}} t^3\right)\right] \tag{5.10-6}$$

where

p_{d} is defined using **PDEC**,

p_{d1} is defined using **PDEC1**, and

p_{d2} is defined using **PDEC2**.

This option is selected by setting **NPRES** = 0.

For both normalized pump head options, the time derivatives of the inlet and exit plenum pressures are calculated as

$$\frac{\mathrm{dp_x}}{\mathrm{dt}} = 0.0 \tag{5.10-7}$$

and

$$\frac{\mathrm{dp_{in}}}{\mathrm{dt}} = \frac{p_2 - p_1}{\Delta t} \tag{5.10-8}$$

where p_1 and p_2 are the pump head values at the beginning and end of the time interval Δt.

5.10.2 Defined Flow Rate

When **IFLOW** $\neq 0$, the inlet pressure is based on flow rate through channel **IFLOW**. The inlet pressure $p_{\mathrm{in}}(t)$ is calculated as

$$p_{\mathrm{in}}(t) = p_{\mathrm{x}} + I_1 \frac{dw}{dt} + g\left(I_5 + \left[\rho_{\mathrm{HOT}}\left(z_{\mathrm{PU}} - z_{\mathrm{MZC}}\right) + \rho_{\mathrm{COLD}}\left(z_1 - z_{\mathrm{PL}}\right)\right]\right) + \Delta P_w \tag{5.10-9}$$

where

$$\Delta P_w = w_1^2 I_2 + A_{\mathrm{fr}} w_1 \left|w_1\right|^{1+b_{\mathrm{fr}}} I_3 + w_1 \left|w_1\right| I_4 \tag{5.10-10}$$

I_1, I_2, I_3, I_4, I_5, A_{fr}, and b_{fr} are the momentum equation contributions, defined in Section 3.9.2, for channel **IFLOW**,

z_1, and z_{MZC} are the elevations at the bottom and top of the fuel channel, and

ρ_{HOT}, and ρ_{COLD} are the coolant density in the upper and lower plenum at time t.

The channel flow rate and its derivative are calculated as

$$w_1 = w_0 f(t) \tag{5.10-11}$$

and

$$\frac{dw}{dt} = w_0 \frac{f\left(t + \Delta t\right) - f\left(t\right)}{\Delta t} \tag{5.10-12}$$

where w_0 is the flow rate through channel **IFLOW** at time $t = 0$, and $f\left(t\right)$ is the user-supplied time dependence of the normalized flow rate. $f\left(t\right)$ is defined in **PRETAB** and **PRETME** or a *function block*, depending on the value of **NPRES**.

The time derivatives of the inlet and exit plenum pressures are calculated as

$$\frac{\mathrm{dp_x}}{\mathrm{dt}} = 0.0 \tag{5.10-13}$$

and

$$\frac{\mathrm{dp_{in}}}{\mathrm{dt}} = \frac{\Delta P_w}{\Delta t}\left(\frac{f(t+\Delta t)}{f(t)}\left|\frac{f(t+\Delta t)}{f(t)}\right|^{F_{w2}} - 1\right) \tag{5.10-14}$$

where

$$F_{w2} = \frac{\partial \Delta P_w}{\partial w} \frac{w_1}{\Delta P_w} - 1 \tag{5.10-15}$$

$$\frac{\partial \Delta P_w}{\partial w} = 2 w_1 I_2 + A_{\text{fr}} \left(2 + b_{\text{fr}}\right) |w_1|^{1+b_{\text{fr}}} I_3 + 2 |w_1| I_4 \tag{5.10-16}$$

5.11 Interaction With Other Models

5.11.1 Information Received by PRIMAR from Other Models

PRIMAR-1 receives no information from other routines. It is driven entirely by the input regardless of what happens elsewhere in the code. Three kinds of input can be supplied: a pressure drop as a function of time, a normalized channel flow rate as a function of time, or the parameters for an exponentially decreasing pressure drop. These are described in Section 5.10 and mentioned in the input listing in Chapter 3.

PRIMAR-4 receives C_0', C_1, C_2, and C_3' and also the inlet and outlet temperatures for each core channel from the coolant dynamics routines. These are the variables described in Section 5.2.2. The coolant dynamics routines also supply PRIMAR with the net mass flow and the net mass flow times temperature from all of the core channels to each outlet plenum, as well as the net mass flow and the net mass flow times temperature into all of the core channels from each inlet plenum during the last PRIMAR time step. The PRIMAR model then adjusts the inlet and outlet plenum mass, pressure, and cover-gas interface to account for differences between the estimated channel flows (PRIMAR) and the computed channel flows (coolant dynamics routines) for the last PRIMAR time step. Symbolically,

$$\Delta m = \int w_{\text{c}} \text{dt} - \int w_{\text{e}} \text{dt} \tag{5.11-1}$$

where

Δm = the mass adjustment

w_{c} = the total calculated mass flow rate from all the core channels

w_{e} = the total estimated mass flow rate from all the core channels

The integration is over the time step. Then

$m_{\text{l}} = m + \Delta m$

$V_{\text{l}} = m_{\text{l}} / \rho_{\text{l}}$

$z_{\text{inter}} = z_{\text{ref}} + V_{\text{l}} / A_{\text{inter}}$

where

m_{l} = the adjusted liquid mass in the plenum

m = the liquid mass in the plenum before adjustment

V_{l} = the liquid volume in the plenum

ρ_{l} = the liquid density in the plenum

z_{inter} = the liquid interface height in the plenum

z_{ref} = the reference height input for the plenum

A_{inter} = the area of the liquid interface input for the plenum

The coefficients C_0', C_1, C_2, and C_3', supplied to the PRIMAR model by the coolant dynamics routines, are used to estimate the rate of change in the mass flow rate for each channel, as indicated in Eq. (5.2-21) and written more simply for a particular channel as

$$\frac{\mathrm{dw}}{\mathrm{dt}} = C_0' + C_1\left(p_{in} + \theta_2 \Delta p_{in}\right) + C_2\left(p_{\mathrm{x}} + \theta_2 \Delta p_{\mathrm{x}}\right) + 2\theta_2 C_3' w \Delta w \tag{5.11-2}$$

where the symbols have the same meaning as in Eq. (5.2-26). Before boiling begins in a core channel, dw/dt depends on $p_{\mathrm{in}} - p_{\mathrm{x}}$, and C_1 is set equal to $-C_2$. After boiling starts, the inlet flow is independent of p_{x}, and depends on $p_{\mathrm{in}} - p_{\mathrm{b}}$, where p_{b} is the bubble pressure. Also after boiling starts, the outlet flow is independent of p_{in}, and depends on $p_{\mathrm{b}} - p_{\mathrm{x}}$. The bubble pressure p_{b} is included in C_0'.

5.11.2 Information Supplied by PRIMAR to Other Models

PRIMAR supplies information to the pre-voiding coolant dynamics routines, the boiling model, LEVITATE and PLUTO-2. Both PRIMAR-1 and PRIMAR-4 supply the same information, but it is arrived at differently by the two options. The information supplied is

$p_{\mathrm{in}}\left(t_{\mathrm{P1}}\right)$ = the inlet plenum pressure at the beginning of the PRIMAR time step

$p_{\mathrm{out}}\left(t_{\mathrm{P1}}\right)$ = the outlet plenum pressure at the beginning of the PRIMAR time step

$\frac{\mathrm{dp_{in}}}{\mathrm{dt}}$, $\frac{\mathrm{dp_{out}}}{\mathrm{dt}}$ = the time derivatives of the inlet and outlet plenum pressures

$T_{\mathrm{in}} T_{\mathrm{out}}$ = the inlet and outlet plenum temperatures

At any time t during the PRIMAR time step, the inlet and outlet plenum pressures are taken as

$$p_{\mathrm{in}}(t) = p_{\mathrm{in}}\left(t_{\mathrm{P1}}\right) + \left(t - t_{\mathrm{P1}}\right)\mathrm{dp_{in}}/\mathrm{dt} \tag{5.11-3}$$

and

$$p_{\mathrm{out}}(t) = p_{\mathrm{out}}\left(t_{\mathrm{P1}}\right) + \left(t - t_{\mathrm{P1}}\right)\mathrm{dp_{out}}/\mathrm{dt} \tag{5.11-4}$$

The plenum pressures p_{in} and p_{out} that PRIMAR supplies to the coolant routines are at the plenum reference heights z_{PLENL} and z_{PLENU}, and the coolant routines compute the core channel inlet and outlet pressures from these by the formula:

$$p\left(z_{\mathrm{ci}}\right) = p_{\mathrm{in}} + \rho_{\mathrm{in}} g\left(z_{\mathrm{PLENL}} - z_{\mathrm{ci}}\right) \tag{5.11-5}$$

and

$$p\left(z_{\mathrm{co}}\right) = p_{\mathrm{out}} + p_{\mathrm{out}} g\left(z_{\mathrm{PLENU}} - z_{\mathrm{co}}\right) \tag{5.11-6}$$

It should be noticed that in the PRIMAR-4 input (**ZCVL**) a reference height is entered for each compressible volume, including the compressible volumes that are the inlet and outlet plenums. In addition, inlet and outlet plenum reference heights, z_{PLENL} and z_{PLENU}, are also entered as input (**ZPLENL** and **ZPLENU**, or **ZPLENC**). If these two sets of reference are different, the PRIMAR-4 code calculates the appropriate adjustments and passes the inlet and outlet plenum pressures at z_{PLENL} and z_{PLENU} to the coolant routines.

The inlet and outlet plenum temperatures are discussed in Section 3.3.6 of Chapter 3.

5.12 Subroutine Descriptions and Flow Charts

5.12.1 Subroutine Listing and Description

The subroutines used in the PRIMAR-4 module are listed in Table 5.12.1. The driver for the transient subroutines is PRIMAR, and the driver for the steady-state subroutines is SSPRM4.

5.12.2 PRIMAR-4 Module Flow Chart

The main subroutines in the PRIMAR-4 module are linked as shown in Figure 5.12.1.

Table 5.12.1: PRIMAR-4 Subroutines

Routine	Description
ANETMP	Annular element temperatures
AVGVAL	Average motor torque over a time step for centrifugal pump
BYPSTM	Temperatures of the coolant, reflectors, and duct wall for a bypass channel; called after the flow is known
CKVLFL	Check valve flow coefficients
COREFL	Estimates core channel flow
CPCPHT	Component to component heat transfer
CUTBAK	Re-initializes variables when the PRIMAR time step is cut back
CVTEMP	Liquid temperatures for a compressible volume, using a one-point perfect mixing model
DRACSF	Shell-side flow coefficients for DRACS heat exchanger
DRACTF	Tube-side flow coefficients for DRACS heat exchanger
GRVHED	Gravity head
IHXSHF	Flow coefficients for IHX shell-side
IHXTBF	Flow coefficients for IHX tube-side
INCPVF	Pressure coefficients for an almost incompressible liquid volume with no cover gas
INITST	Initializes arrays for a sub-interval
INPLNF	Inlet plenum pressure coefficients
LIQCV	Pressure coefficients for a compressible volume with no cover gas
LIQFIN	Compressible volume pressure at end of sub-interval, with liquid only and no cover gas
LQGSFN	Compressible volume pressure at end of sub-interval, with liquid plus cover gas
OUTNGF	Pressure coefficients for a compressible outlet plenum, no cover gas
OUTPLF	Pressure coefficients for an outlet plenum with cover gas
PIPEFL	Pressure coefficients for pipe flow with friction
PIPTMP	Pipe temperatures using Lagrangian mesh
PMPBLF	Pressure coefficients for a pump bowl with cover gas
PMSTRT	Initializes arrays for a sub-interval
POOLFL	Pressure coefficients for a pool
PRESDR	Pressure drop in a flow element
PRIMAR	Main driver for the module
PRIMR1	Simple PRIMAR-1 option, with user-specified driving head vs. time, or flow vs. time
PRMADJ	Adjusts inlet and outlet plenum variables to account for the differences between estimated and computed channel flows

continues on next page

Table 5.12.1 – continued from previous page

Routine	Description
PRMEND	Puts final sub-interval results into permanent arrays for storage
PRMPRT	Prints PRIMAR-4 results
PRSRZF	Pressure coefficients for a pressurized (expansion tank)
PRSTEP	Finds the next PRIMAR time-step size or decides to cut back the current step size
PUMPFL	Pump head and flow coefficients
PUMPFN	Pump impeller speed and head calculation at the end of a sub-interval
RUPSKF	Pressure coefficients for a pipe rupture sink, guard vessel with cover gas
RUPSRF	Pipe rupture source pressure coefficients
SELECT	Chooses average and final inlet/outlet temperature into/from a compressible volume, according to segment flow directions
SSADHX	Steady-state air dump heat exchanger
SSBYPS	Bypass channel steady-state coolant and wall temperatures, given steady-state flow rate and steady-state power
SSCKVL	Steady-state check valve pressure coefficents
SSCPNL	Steady state null transient driver
SSIHX	Steady-state IHX temperatures
SSIHXC	Steady-state IHX temperatures, simple model
SSLQSG	Steady-state initialization of the elements in a liquid segment
SSPMLP	Steady-state initialization of primary loop flows, pressures, and temperatures
SSPRM4	Steady-state initialization of PRIMAR-4 variables – driver for the steady-state routines
SSPRPL	Initilization for file 15 binary output
SSPRSR	Steady-state pressure and pump head for one segment of an intermediate loop
SSPUMP	Steady-state initialization of pump parameters
SSP4PR	Steady-state PRIMAR-4 printout
SSP4TH	Initialization of node volumes, VOLNDT(ITGP), coolant and wall temperatures, and liquid and wall temperature arrays for pipe-type temperature groups
SSSCLP	Steady-state initializaton of the intermediate sodium loops, and also calls the steam generator initialization routines
SSSTDR	Steady-state steam generator driver
SSSTGN	Steady-state steam generator initialization of simple table look-up option
SSVALV	Steady-state valve pressures
STEPFN	Finishes liquid flows and temperatures for a sub-interval
STEPGS	Gas flow between compressible volumes and storage tanks for a sub-interval
STEPLQ	Driver for liquid flow and pressure calculations
STEPTM	Driver for liquid temperature calculations
STGNFL	Steam generator flow coefficients, sodium side
STRATV	Stratified volume calculations
STRGVH	Calculates gravity heads in a stratified volume
SUBSIZ	Sets the PRIMAR time sub-interval size
TSDRCS	DRACS temperatures
TSIHX	IHX temperatures, shell and tube sides, and gravity head
TSIHXC	IHX temperatures and gravity head, simple model
TSPRPL	Writes binary arrays to file 15 for later use
TSSTGN	Steam generator temperatures and gravity head, simple model
VALVAJ	Steady-state valve pres. drop adjustment. If needed, between compressible volumes connected by several liquid segments
VALVFL	Valve pressure coefficients

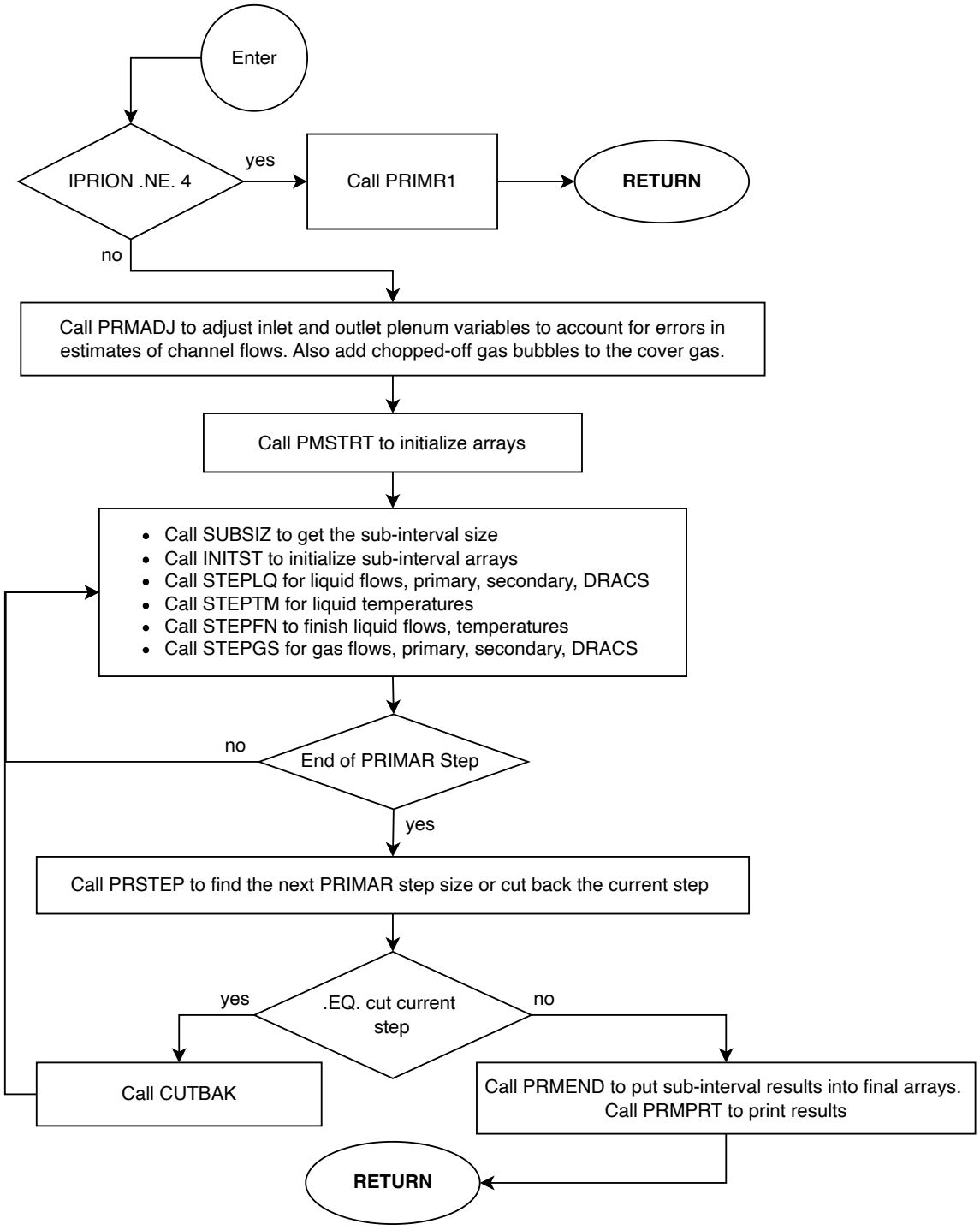


Figure 5.12.1: PRIMAR-4 Module Flow Chart.

5.13 Input/Output Description

5.13.1 Input Description

The input description can be divided into the input needed for the PRIMAR-1 option and the input needed for the PRIMAR-4 option.

5.13.1.1 PRIMAR-1 Option

For the PRIMAR-1 option two variables need to be supplied for INPCOM (Block 1) and all the variables for PRIMIN (Block 14). The two variables for INPCOM are **NPRES** and **IPRION**.

5.13.1.2 PRIMAR-4 Option

For the PRIMAR-4 option, one variable needs to be supplied for INPCOM (Block 1), eight variables for PRIMIN (Block 14), and all of the variables for INPMR4 (Block 3) and PMR4IN (Block 18) that are appropriate for the model being considered. The one variable for INPCOM is **IPRION**. The eight variables for PRIMIN are

Variable Name	Location
PX	1
ZPLENL	87
ZPLENU	88
DZBCGL	90
DZBCGU	91
XMXMSI	93
XMXMSO	94
TIMMIX	95

INPMR4 contains the integer input for PRIMAR-4 and PMR4IN contains the floating-point input, as described in the input listing in Chapter 2. Only input information pertinent to the model being calculated needs to be entered. For example, if no intermediate loop is present, no information for the intermediate loop in either INPMR4 or PMR4IN need be entered. Any variable not entered is assumed to be zero, since all of the input blocks are zeroed out before any input information is read in. Also, any extraneous information, which may be left over from an old input deck is ignored, provided all entries are made correctly. That is, if the number of compressible volumes in an intermediate loop is entered as zero, then any compressible volume information supplied for that intermediate loop will be ignored.

An example of INPMR4 and PMR4IN input, the integer and floating-point input blocks for PRIMAR-4, is shown below for the example used in Section 5.13.2.

```
INPMR4      3      0      0
      1     10      5      0      0      5      0      0      3      0      0     14
     11     10      1      7      9     11     11      0      0      0      0      0
     41     10      0      0      0      0      0      0      0      0      1      3
     51     10      2      3      3      2      3      3      5      3      6      3
     61     10      3      3      0      0      0      0      0      0      0      0
    181     10      0      0      0      0      0      0      0      0      1      2
    191     10      1      2      1      2      2      3      3      1      0      0
    261     10      0      0      0      0      0      0      0      0      2      4
```

(continues on next page)

(continued from previous page)

```
   271    10     3     4     4     5     0     0     0     0     0     0
   321    10     0     0     0     0     1     3     3     1     6     0
   361    10     0     0     0     0     1     2     5     8     9     0
   401    10     0     0     0     0     1     9     0     0     0     0
   411    10     0     0     0     0     0     0     0     1     0     0
   461    10     0     0     0     0     0     0     0     0     0     1
   471    10     0     0    11     0     0     0     0     0     0     0
   481    10     0     0     0     0     0     0     0     0     1     0
   511    10     0    10     4     7     4     4     7     4    20    16
   521    10    41    30     0     0     0     0     0     0     0     0
   611    10     0     0     2     3     4     5     6     7     8     9
   621    10    11    12     0     0     0     0     0     0     0     0
   711    10     0     0     2     3     4     5     6     7     8    10
   721    10    11    14     0     0     0     0     0     0     0     0
   811    10     0     0     2     7     7     0     0     0     0     0
   821    10     0     1     1     0     0     0     0     0     0     3
   831    10     6     0     0     0     0     0     0     0     0     0
   881    10     0     0     0     0     0     0     0     0     0    50
END
```

```
PMR4IN    18     0     0
     1     5        30.0 4.78500E+03       129.0       303.0 1.73900E+03
     6     5 1.73900E+03 1.73900E+03 1.73900E+03 1.60000E+03 1.60000E+03
    11     5         0.0 1.60000E+03 1.60000E+03         0.0         0.0
    41     5         0.0-2.99600E+00-2.99600E+00-2.99600E+00 4.75600E+00
    46     5 4.83600E+00 4.75600E+00 4.83600E+00 9.52880E+00 1.14340E+01
    51     5 1.14340E+01 9.52880E+00 1.14340E+01 1.14340E+01         0.0
    81     5         0.0         1.0         1.0         1.0         1.0
    86     5         1.0         1.0         3.0         1.0         1.0
    91     5         3.0         2.0         1.0         1.0         2.0
    96     5         1.0         1.0         1.0         1.0         1.0
   101     5         1.0         1.0         1.0         1.0         1.0
   106     5         1.0         1.0         0.0         0.0         0.0
   161     5         0.0         3.0 1.45000E-01         1.1         2.7
   166     5 1.45000E-01         1.1         2.7 4.83600E+00 4.83600E+00
   171     5 9.32580E+00 1.79430E+00-9.00000E-01-9.00000E-01-2.98950E+00
   176     5 4.83600E+00 4.83600E+00 9.32580E+00 1.79430E+00-9.00000E-01
   181     5-9.00000E-01-2.98950E+00 9.52900E+00 1.79430E+00 9.32580E+00
   186     5 1.62920E+01 4.17600E+00 1.14340E+01 1.14340E+01 1.37520E+01
   191     5 9.52900E+00 1.79430E+00 9.32580E+00 1.62920E+01 4.17600E+00
   196     5 1.14340E+01 1.14340E+01 1.37520E+01         0.0         0.0
   301     5         0.0 5.99600E+00 3.14100E+00 9.55000E-01         1.6
   306     5 3.14100E+00 9.55000E-01         1.6 3.35280E+01         1.0
   311     5 2.43840E+01 8.82000E+00 2.23600E+01         1.0 2.23600E+01
   316     5 3.35280E+01         1.0 2.43840E+01 8.82000E+00 2.23600E+01
   321     5         1.0 2.23600E+01         1.0        95.0 8.82000E+00
   326     5       311.0 1.21160E+01       132.0         2.0 1.52400E+01
   331     5         1.0        95.0 8.82000E+00       311.0 1.21160E+01
   336     5       132.0         2.0 1.52400E+01         0.0         0.0
   441     5         0.0         1.0 1.00000E-02 1.00000E-02 1.00000E-02
   446     5 1.00000E-02 1.00000E-02 1.00000E-02 6.20720E-01 2.68000E-01
   451     5 2.68000E-01 1.93600E+00 2.68000E-01 2.68000E-01 2.68000E-01
   456     5 6.20720E-01 2.68000E-01 2.68000E-01 1.93600E+00 2.68000E-01
   461     5 2.68000E-01 2.68000E-01 2.68000E-01 2.68000E-01 8.75000E-01
   466     5 2.68000E-01 2.68000E-01 2.68000E-01 2.68000E-01 2.68000E-01
```

(continues on next page)

(continued from previous page)

```
  471     5 2.68000E-01 2.68000E-01 8.75000E-01 2.68000E-01 2.68000E-01
  476     5 2.68000E-01 2.68000E-01 2.68000E-01         0.0         0.0
  581     5         0.0         1.0 1.00000E-01 1.00000E-01 1.00000E-01
  586     5 1.00000E-01 1.00000E-01 1.00000E-01 8.89000E-01 5.84200E-01
  591     5 5.84200E-01 3.77000E-02 5.84200E-01 5.84200E-01 5.42000E-01
  596     5 8.89000E-01 5.84200E-01 5.84200E-01 3.77000E-02 5.84200E-01
  601     5 5.84200E-01 5.84200E-01 5.84200E-01 5.84200E-01 1.97660E-02
  606     5 5.84200E-01 1.84000E-02 5.84200E-01 5.84200E-01 8.99000E-02
  611     5 5.84200E-01 5.84200E-01 1.97660E-02 5.84200E-01 1.84000E-02
  616     5 5.84200E-01 5.84200E-01 8.99000E-02         0.0         0.0
  721     5         0.0 5.00000E-05 5.00000E-05 5.00000E-05 5.00000E-05
  726     5 5.00000E-05 5.00000E-05 5.00000E-05 5.00000E-05 5.00000E-05
  731     5 5.00000E-05 5.00000E-05 5.00000E-05 5.00000E-05 5.00000E-05
  736     5 5.00000E-05 5.00000E-05 5.00000E-05 5.00000E-05 5.00000E-05
  741     5 5.00000E-05 5.00000E-05 5.00000E-05 5.00000E-05 5.00000E-05
  746     5 5.00000E-05 5.00000E-05 5.00000E-05 5.00000E-05 5.00000E-05
  751     5 5.00000E-05 5.00000E-05 5.00000E-05 5.00000E-05 5.00000E-05
  756     5 5.00000E-05 5.00000E-05 5.00000E-05         0.0         0.0
  861     5         0.0         0.0         1.0         0.0         1.0
  866     5         1.0         0.0         1.0         1.0         0.0
  871     5        10.9         0.0         6.8         1.0         6.8
  876     5         1.0         0.0        10.9         0.0         6.8
  881     5         1.0         6.8         0.0         9.6         0.0
  886     5        18.9         0.0        28.6         1.0         1.0
  891     5         0.0         9.6         0.0        18.9         0.0
  896     5        28.6         1.0         1.0         0.0         0.0
 1141     5         0.0        30.0 1.09550E+05 1.09550E+05 1.09550E+05
 1146     5 1.09550E+05 1.09550E+05 1.09550E+05 1.09550E+05 1.65490E+05
 1151     5 1.09550E+05 1.09550E+05 1.09550E+05 1.09550E+05 1.09550E+05
 1156     5 1.09550E+05 1.65490E+05 1.09550E+05 1.09550E+05 1.09550E+05
 1161     5 1.09550E+05 1.09550E+05 1.09550E+05 1.09550E+05 1.09550E+05
 1166     5 1.09550E+05 1.09550E+05 1.09550E+05 1.09550E+05 1.09550E+05
 1171     5 8.74070E+03 1.09550E+05 1.09550E+05 1.09550E+05 1.09550E+05
 1176     5 1.09550E+05 1.09550E+05 1.09550E+05 8.74070E+03         0.0
 1281     5         0.0         0.0 1.83530E+05 1.83530E+05 1.83530E+05
 1286     5 1.83530E+05 1.83530E+05 1.83530E+05 1.83530E+05 2.79290E+05
 1291     5 1.83530E+05 1.83530E+05 1.83530E+05 1.83530E+05 1.83530E+05
 1296     5 1.83530E+05 2.79290E+05 1.83530E+05 1.83530E+05 1.83530E+05
 1301     5 1.83530E+05 1.83530E+05 1.83530E+05 1.83530E+05 1.83530E+05
 1306     5 1.83530E+05 1.83530E+05 1.83530E+05 1.83530E+05 1.83530E+05
 1311     5 2.79290E+04 1.83530E+05 1.83530E+05 1.83530E+05 1.83530E+05
 1316     5 1.83530E+05 1.83530E+05 1.83530E+05 2.79290E+04         0.0
 1421     5         0.0         0.0        86.4       130.0        16.0
 1426     5         1.0       100.0         1.0         8.0        40.0
 1431     5         1.0         8.0        40.0         1.0         0.0
 1461     5         0.0 1.00700E+05 1.00700E+05         0.0 1.00700E+05
 1466     5 1.00700E+05 7.17000E+05 7.17000E+05         0.0 7.17000E+05
 1471     5 7.17000E+05         0.0         0.0         0.0         0.0
 1496     5         0.0         0.0         0.0 1.10000E-09         0.0
 1506     5         0.0         0.0         0.0         0.0 1.10000E-09
 1571     5         0.0         0.0         0.0         0.0 -1.6000E+00
 1576     5         3.0 4.83600E+00 1.43600E+01 1.43600E+01 1.43600E+01
 1581     5        10.4 1.37500E+01 1.14300E+01        10.4 1.37500E+01
 1586     5 1.14300E+01         0.0         0.0         0.0         0.0
 1611     5         0.0         0.0         0.0        27.0 3.07000E+00
```

(continues on next page)

(continued from previous page)

```
 1616     5         1.0        10.0         1.0         2.5         6.0
 1621     5         0.0         2.5         6.0         0.0         0.0
 1651     5       660.9         0.0         0.0       808.2       808.2
 1656     5       808.2         0.0         0.0       616.5         0.0
 1661     5         0.0       616.5         0.0         0.0         0.0
 1686     5         0.0         0.0         0.0         1.6       208.1
 1691     5         1.0       100.0        15.0        15.0         5.0
 1696     5         5.0        15.0        15.0         0.0         0.0
 1721     5 3.49150E-03 3.49150E-03 6.65840E-03 6.65840E-03 3.49150E-03
 1726     5 3.49150E-03         0.0         0.0         0.0         0.0
 1746     5         0.0         0.0         0.0         1.0         1.0
 1751     5         1.0         3.0         1.0         1.0         1.0
 1756     5         1.0         1.0         1.0         1.0         1.0
 1801     5         0.0         0.0         0.0         0.0 6.66750E-02
 1806     5 6.66750E-02 9.20750E-02 9.20750E-02 6.66750E-02 6.66750E-02
 1861     5         0.0         1.0         1.0         1.0         1.0
 1866     5         1.0         0.0         0.0         0.0         0.0
 1896     5         0.0         0.0         0.0 1.20000E+03         0.0
 1911     5 1.12675E+06         0.0         0.0         0.0         0.0
 1921     5         0.0         0.0 1.16868E+02         0.0         0.0
 1931     5         0.0         0.0         0.0         0.0 1.93200E+03
 1946     5         0.0 9.00000E-01         0.0         0.0         0.0
 1956     5         0.0         0.0         0.0 1.00000E-01         0.0
 1981     5         0.0         0.0 1.22900E+00-1.80000E-01 1.92000E-01
 1986     5-2.80000E-01 3.90000E-02         3.0-2.20400E-01 1.14400E-01
 1991     5 8.05000E-02 6.02000E-02 5.84000E-01 3.30000E-02 5.53000E-01
 1996     5-1.52000E-01-1.90000E-02 8.00000E-03 8.00000E-03         0.0
 2001     5         0.0 3.24000E-02         1.0         1.0 5.12000E-01
 2006     5 3.79000E-01 2.87000E-01 2.21000E-01 1.74000E-01 1.14400E-01
 2011     5 8.05000E-02 6.02000E-02 4.76000E-02 3.96000E-02 2.64000E-02
 2016     5 1.76000E-02 1.18000E-02 8.00000E-03 8.00000E-03         0.0
 2021     5         0.0 3.24000E-02         1.0         1.0 3.00000E-01
 2026     5 1.90000E-01 1.20000E-01 8.00000E-02 6.00000E-02 4.00000E-02
 2031     5 3.00000E-02 2.00000E-02 1.60000E-02 1.40000E-02 1.20000E-02
 2036     5 1.00000E-02 9.00000E-03 8.00000E-03 8.00000E-03         0.0
 2041     5         0.0 3.24000E-02         1.0         1.0 3.00000E-01
 2046     5 1.90000E-01 1.20000E-01 8.00000E-02 6.00000E-02 4.00000E-02
 2051     5 3.00000E-02 2.00000E-02 1.60000E-02 1.40000E-02 1.20000E-02
 2056     5 1.00000E-02 9.00000E-03 8.00000E-03 8.00000E-03         0.0
 2061     5         0.0 3.24000E-02         0.0         0.0         0.0
 2221     5         0.0         0.0         1.0         0.0         0.0
 2226     5         3.0         4.0         5.0         6.0         8.0
 2231     5        10.0        12.0         0.0 1.00000E-02 1.00000E+03
 2236     5        24.0        28.0        32.0       400.0       401.0
 2241     5 1.00000E+04        19.0         0.0 1.00000E+03         2.0
 2246     5         3.0         4.0         5.0         6.0         8.0
 2251     5        10.0        12.0        14.0        16.0        20.0
 2256     5        24.0        28.0        32.0       400.0       401.0
 2261     5 1.00000E+04        19.0         0.0 1.00000E+03         2.0
 2266     5         3.0         4.0         5.0         6.0         8.0
 2271     5        10.0        12.0        14.0        16.0        20.0
 2276     5        24.0        28.0        32.0        33.0        40.0
 2281     5 1.00000E+04        19.0         0.0 1.00000E+03         2.0
 2286     5         3.0         4.0         5.0         6.0         8.0
 2291     5        10.0        12.0        14.0        16.0        20.0
```

(continues on next page)

(continued from previous page)

```
 2296     5        24.0        28.0        32.0        33.0        40.0
 2301     5 1.00000E+04        19.0         0.0         0.0         0.0
 2461     5         0.0         0.0         9.8 2.13000E-10 2.13000E-10
 2466     5 2.13000E-10         0.0         0.0         0.0 2.13000E-10
 2471     5 2.13000E-10 2.13000E-10 2.13000E-10 2.13000E-10 2.13000E-10
 2501     5         0.0-2.80000E-04-2.80000E-04-2.80000E-04-2.80000E-04
 2506     5-2.80000E-04-2.80000E-04-2.80000E-04-2.80000E-04-2.80000E-04
 2511     5-2.80000E-04-2.80000E-04-2.80000E-04         0.0         0.0
 2536     5         0.0         0.0         0.0         0.0       844.0
 2541     5       844.0       844.0       844.0       844.0       844.0
 2546     5       844.0       844.0       844.0       844.0       844.0
 2551     5       844.0         0.0         0.0         0.0         0.0
 2576     5         0.0         0.0 5.00000E+04 1.00000E+03 5.00000E+04
 2581     5 5.00000E+04 5.00000E+04 5.00000E+04 5.00000E+04 5.00000E+04
 2586     5 5.00000E+04 5.00000E+04 5.00000E+04 5.00000E+04         0.0
 2616     5        19.8        39.3         6.4         1.0        21.8
 2621     5         1.0         4.0        11.8         1.0         4.0
 2626     5        11.8         1.0         0.0         0.0         0.0
 2651     5         0.0         0.0         0.0 7.59000E+06 1.30000E+08
 2656     5 2.45000E+06 3.83000E+05 8.35000E+06 3.83000E+05 1.53000E+06
 2661     5 4.52000E+06 3.83000E+05 1.53000E+06 4.52000E+06 3.83000E+05
 2691     5         0.0         0.0         0.0 1.60000E-02 8.60000E-01
 2696     5 4.55000E+00 1.96200E+01 1.96200E+01         0.0         0.0
 2701     5         0.0         0.0         0.0         0.0 1.96200E+01
 2706     5 1.96200E+01         0.0         0.0         0.0         0.0
 2711     5         0.0         0.0 1.96200E+01 1.96200E+01         0.0
 2721     5 1.96200E+01 1.96200E+01         0.0         0.0         0.0
 2726     5         0.0         0.0         0.0 1.96200E+01 1.96200E+01
 2736     5         0.0 6.13000E-04 6.13000E-04         0.0         0.0
 2741     5         0.0         0.0         0.0         0.0 6.13000E-04
 2746     5 6.13000E-04         0.0         0.0         0.0         0.0
 2751     5         0.0         0.0 6.13000E-04 6.13000E-04         0.0
 2761     5 6.90000E+06 6.90000E+06         0.0         0.0         0.0
 2766     5         0.0         0.0         0.0 6.90000E+06 6.90000E+06
 2776     5         0.0 6.90000E+06 6.90000E+06         0.0         0.0
 2781     5         0.0         0.0         0.0         0.0 6.90000E+06
 2786     5 6.90000E+06         0.0         0.0         0.0         0.0
 2791     5         0.0         0.0 6.90000E+06 6.90000E+06         0.0
 2801     5         1.0         1.0         1.0         1.0         1.0
 2806     5         1.0         1.0         1.0         1.0         1.0
 2811     5         1.0         1.0         1.0         1.0         0.0
 2856     5         0.0 9.77550E-04         0.0         0.0         0.0
 2861     5         0.0 9.77550E-04         0.0         0.0         0.0
 2921     5 1.83000E-02 1.83000E-02         0.0         0.0         0.0
 2926     5         0.0         0.0         0.0 1.83000E-02 1.83000E-02
 2936     5         0.0         1.0         1.0         0.0         0.0
 2951     5         1.0         1.0         0.0         0.0         0.0
 3101     5         0.0         0.0         0.0         0.0 5.51000E+00
 3106     5 5.51000E+00        10.5        10.5        10.5         0.0
 3116     5         0.0         0.0         0.0        10.5        10.5
 3121     5        10.5        10.5        10.5         0.0         0.0
 3271     5         0.0         0.0         0.0 1.00000E+03        15.0
 3276     5 1.00000E+03 1.00000E+03         0.0         0.0         0.0
 3286     5         0.0         0.0 1.00000E+03        15.0 1.00000E+03
 3291     5 1.00000E+03         0.0         0.0         0.0         0.0
```

(continues on next page)

(continued from previous page)

```
  3441     5          0.0          0.0          0.0          0.0 1.80000E+06
  3446     5 1.80000E+06          0.0          0.0 1.60000E-02 8.60000E-01
  3451     5 4.55000E+00 1.60000E-02 8.60000E-01 4.55000E+00          1.0
  3456     5          1.0          1.0          1.0          1.0          1.0
  3461     5          1.0          1.0          1.0          1.0          1.0
  3466     5          1.0          1.0          1.0          1.0          1.0
  3471     5          1.0          1.0          1.0          1.0          1.0
  3476     5          1.0          1.0          1.0          1.0          1.0
  3481     5          1.0          1.0          1.0          1.0          1.0
  3486     5          1.0          1.0          1.0          1.0          1.0
  3491     5          1.0          1.0          1.0          1.0          1.0
  3496     5          1.0          1.0          1.0          1.0          1.0
  3501     5          1.0          1.0          1.0          1.0          1.0
  3506     5          1.0          1.0          1.0          1.0          1.0
  3511     5          1.0          1.0          1.0          1.0          1.0
  3516     5 6.56850E+00 6.56850E+00          0.0          0.0          0.0
  3521     5          0.0          0.0          0.0 1.98990E+02 1.98990E+02
  3531     5          0.0 1.76990E+02 1.76990E+02          0.0          0.0
  3536     5          0.0          0.0          0.0          0.0 1.09900E-01
  3541     5 1.09900E-01          0.0          0.0          0.0          0.0
  3546     5          0.0          0.0 1.22865E-03 1.22865E-03          0.0
  3556     5 4.47000E+06 4.47000E+06          0.0          0.0          0.0
  3561     5          0.0          0.0          0.0 4.47000E+06 4.47000E+06
  3571     5          0.0 1.96200E+01 1.96200E+01          0.0          0.0
  3576     5          0.0          0.0          0.0          0.0 1.96200E+01
  3581     5 1.96200E+01          0.0          0.0          0.0          0.0
  3586     5          0.0          0.0 1.00000E+08 1.00000E+08          0.0
  3596     5 1.00000E+08 1.00000E+08          0.0          0.0          0.0
  3601     5          0.0          0.0          0.0          1.0          1.0
  3606     5          0.0          0.0          1.0          1.0          0.0
  3611     5          0.0          0.0         60.0          5.6          1.0
  3616     5        100.0          1.0          2.7         23.0          0.0
  3621     5          2.7         23.0          0.0          0.0          0.0
  3646     5          0.0          0.0          0.0          0.0 1.60000E-02
  3651     5 8.60000E-01 4.55000E+00 1.60000E-02 8.60000E-01 4.55000E+00
  3656     5 1.95300E+00          0.0          0.0          0.0          0.0
END
```

5.13.2 Example of a Primary Loop

As an aid to the user in preparing PRIMAR-4 input data, the primary loop in Figure 5.13.1 is described.

This primary loop consists of five compressible volumes, CV1,…,CV5, five liquid segments, S1,…S5, fourteen elements, E1,…,E14, distributed among the liquid segments, nine temperature groups, T1,…,T9, of liquid flow elements, and three gas segments. The compressible volumes are:

CV1 = inlet plenum

CV2 = outlet plenum with cover gas

CV3 = pump bowl and cover gas

CV4 = compressible gas volume, no liquid

CV5 = compressible gas volume, no liquid

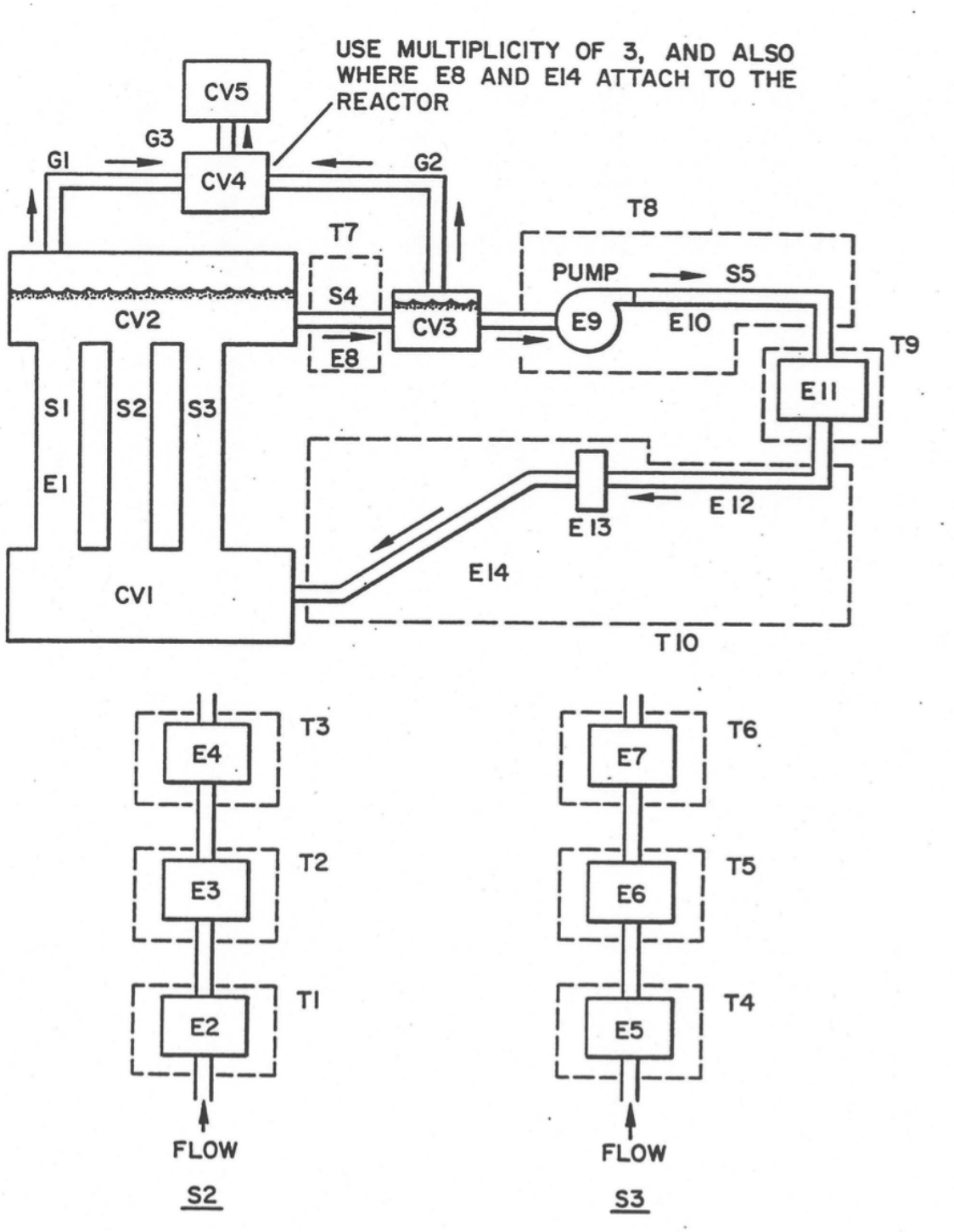


Figure 5.13.1: Example of a Primary Loop.

The elements are:

E1 = core subassemblies

E2, E4, E5, E7, E8, E10, E12, E13, E14 = pipes

E3, E6 = bypass channels

E9 = pump impeller

E11 = shell side of intermediate heat exchanger

The elements in liquid segment S5, for example, are E9, E10, E11, E12, E13, and E14.

The assumed directions of flow for the liquid, as well as for the gas, segments are indicated in the diagram. It is this assumed direction of flow that is used in choosing the inlet and outlet compressible volumes for a liquid or for a gas segment. For example, CV3 is the inlet compressible volume and CV1 is the outlet compressible volume for liquid segment S5, while CV3 and CV4 are the inlet and outlet compressible volumes for gas segment G2. It is also with respect to the assumed directions of the segments that positive and negative flow rates are interpreted. Finally, it should be observed that positive inlet flow to a liquid or gas segment is flow out of the compressible volume at the segment inlet.

The multiplicity of 3 that appears in Figure 5.13.1 where G2 joins CV4, where E8 joins CV2, and where E14 joins CV1 is present because in this example there are actually three identical primary loops attached to the respective compressible volumes, and instead of performing three identical calculations, the code computes one primary loop and multiplies the appropriate flows by 3. It should be noticed, however, that the multiplicity for liquid segment S4 is 3 at the inlet and 1 at the outlet, for liquid segment S5 it is 1 at the inlet and 3 at the outlet, and for gas segment G2 it is 1 at the inlet and 3 at the outlet. The multiplicity for all other liquid and gas segments must be set to 1 at both ends.

The liquid temperature calculations, which are performed after the liquid flow calculations have been completed, are done by temperature groups. This allows for the grouping of adjacent elements whose temperatures are computed in a similar way, such as elements E12, E13, and E14, all of which are pipes in temperature group T10, and also elements E9, and E10, a pump and connecting pipe, in temperature group T8. Where such grouping is inappropriate, a single element comprises a temperature group, such as the bypass channels as T2 and T5 and the intermediate heat exchanger as T9.

Although not shown in the figure, if an intermediate loop were to be added, it would be structured in a fashion similar to the primary loop, and it would contain an element representing the tube side of the intermediate heat exchanger and would be located next to E11 in the diagram.

5.13.3 Output Description

The PRIMAR output information is largely self-explanatory. Specimens of three output printouts for the example in Section 5.13.2 are displayed. The first is an input edit in which all of the input variables in Blocks 3 and 18 are echoed back. The second is a print of the steady-state values calculated by the PRIMAR steady-state routines. Finally, the third is the print of the values calculated by the transient routines at a later time step.

SAS4A 0.0 OUTLET PLENUM WITH COVER GAS 2/07/84 03.22.44

PRIMAR-4 INPUT EDIT

INTEGER CONSTANTS

NCVP	NCVS	NCVD	NSEGLP	NSEGLS	NSEGLD	NSEGGP	NSEGGS	NSEGGD	NELEMT	NPUMP	NIHX	NDRACS	NCKV	IDBPR4	IDBP4N
5	0	0	5	0	0	3	0	0	14	1	1	0	0	0	0

INAKDR	ISRCRP	ISNKRP	NTGPT	NBYP	NSGN	IP4PRT	NBNTOT	IBINST
0	0	0	10	2	0	50	-1	0

FLOATING-POINT CONSTANTS

TMDSP4 S	BHDLOD	GAMGSC	RGASC PA-M**3/KG-K	U0CVGS PA-S	TRFU K	GRAVTY M/S/S	VOLMIX M**3
3.000000D+01	3.000000D+01	1.600000D+00	2.081000D+02	1.000000D+00	1.000000D+02	9.800000D+00	0.0

TAUUP S	C1BY	C2BY	C3BY	C1PIPE	C2PIPE	C3PIPE	C1IHX
0.0	1.600000D-02	8.600000D-01	4.550000D+00	1.600000D-02	8.600000D-01	4.550000D+00	1.600000D-02

C2IHX,	C3IHX
8.600000D-01	4.550000D+00

LIQUID FLOW ELEMENT TYPE

ELEMENT	TYPE	DESCRIPTION	ZOUTEL M	XLENEL M
1	1	CORE SUBASSEMBLIES	3.000000D+00	5.996000D+00
2	3	PIPE	1.450000D-01	3.141000D+00
3	2	BYPASS CHANNEL	1.100000D+00	9.550000D-01
4	3	PIPE	2.700000D+00	1.600000D+00
5	3	PIPE	1.450000D-01	3.141000D+00
6	2	BYPASS CHANNEL	1.100000D+00	9.550000D-01
7	3	PIPE	2.700000D+00	1.600000D+00
8	3	PIPE	4.836000D+00	3.352800D+01
9	5	PUMP IMPELLER	4.836000D+00	1.000000D+00
10	3	PIPE	9.325800D+00	2.438400D+01
11	6	IHX, PRIMARY SIDE	1.794300D+00	8.820000D+00
12	3	PIPE	-9.000000D-01	2.236000D+01
13	3	PIPE	-9.000000D-01	1.000000D+00
14	3	PIPE	-2.989500D+00	2.236000D+01

SAS4A 0.0 OUTLET PLENUM WITH COVER GAS 2/07/84 03.22.44

ELEMENT	AREAEL M**2	DIAELEM M	ROUGHL M	BENDNM	G2PROR	WALLMC J/M/K	WALLH J/M/K
1	1.0000000D+00	1.0000000D+00	5.0000000D-05	0.0	0.0	1.0955000D+05	1.8353000D+05
2	1.0000000D-02	1.0000000D-01	5.0000000D-05	1.0000000D+00	0.0	1.0955000D+05	1.8353000D+05
3	1.0000000D-02	1.0000000D-01	5.0000000D-05	0.0	0.0	1.0955000D+05	1.8353000D+05
4	1.0000000D-02	1.0000000D-01	5.0000000D-05	1.0000000D+00	0.0	1.0955000D+05	1.8353000D+05
5	1.0000000D-02	1.0000000D-01	5.0000000D-05	1.0000000D+00	0.0	1.0955000D+05	1.8353000D+05
6	1.0000000D-02	1.0000000D-01	5.0000000D-05	0.0	0.0	1.0955000D+05	1.8353000D+05
7	1.0000000D-02	1.0000000D-01	5.0000000D-05	1.0000000D+00	0.0	1.0955000D+05	1.8353000D+05
8	6.2072000D-01	8.8900000D-01	5.0000000D-05	1.0000000D+00	0.0	1.6549000D+05	2.7929000D+05
9	2.6800000D-01	5.8420000D-01	5.0000000D-05	0.0	0.0	1.0955000D+05	1.8353000D+05
10	2.6800000D-01	5.8420000D-01	5.0000000D-05	1.0900000D+01	0.0	1.0955000D+05	1.8353000D+05
11	1.9360000D+00	3.7700000D-02	5.0000000D-05	0.0	0.0	1.0955000D+05	1.8353000D+05
12	2.6800000D-01	5.8420000D-01	5.0000000D-05	6.8000000D+00	0.0	1.0955000D+05	1.8353000D+05
13	2.6800000D-01	5.8420000D-01	5.0000000D-05	1.0000000D+00	0.0	1.0955000D+05	1.8353000D+05
14	2.6800000D-01	5.8420000D-01	5.0000000D-05	6.8000000D+00	0.0	1.0955000D+05	1.8353000D+05

COMPRESSIBLE VOLUMES IN PRIMARY LOOPS

VOLUME	TYPE	DESCTIPTION	VOLLGC M**3
1	1	INLET PLENUM	8.6400000D+01
2	7	OUTLET PLENUM WITH COVER GAS	1.3000000D+02
3	9	PUMP BOWL AND COVER GAS	1.6000000D+01
4	11	COMPRESSIBLE GAS VOLUME, NO LIQUID	1.0000000D+00
5	11	COMPRESSIBLE GAS VOLUME, NO LIQUID	1.0000000D+02

VOLUME	PRESGO PA	ALPHAP 1/PA	ALPHAT 1/K	ZCVL M	AREAIN M**2	TREFCV K	TAUGAS S
1	0.0	1.1000000D-09	0.0	-1.6000000D+00	0.0	6.6090000D+02	0.0
2	1.0070000D+05	0.0	0.0	3.0000000D+00	2.7000000D+01	0.0	1.0000000D+00
3	1.0070000D+05	0.0	0.0	4.8360000D+00	3.0700000D+00	0.0	1.0000000D+00
4	1.0070000D+05	0.0	0.0	1.4360000D+01	1.0000000D+00	8.0820000D+02	1.0000000D+00
5	1.0070000D+05	0.0	0.0	1.4360000D+01	1.0000000D+01	8.0320000D+02	1.0000000D+00

VOLUME	BTAPHA 1/PA	BTATHA 1/K	RHONAR KG/M**3	HWALL W/M**2/K	AWALL M**2	CMWALL J/K	VOLGSO M**3
1	2.1300000D-10	-2.8000000D-04	8.4400000D+02	5.0000000D+04	1.9800000D+01	7.5900000D+06	0.0
2	2.1300000D-10	-2.8000000D-04	8.4400000D+02	1.0000000D+03	3.9300000D+01	1.3000000D+08	6.0000000D+01
3	2.1300000D-10	-2.8000000D-04	8.4400000D+02	5.0000000D+04	6.4000000D+00	2.4500000D+06	5.6000000D+00
4	0.0	-2.8000000D-04	8.4400000D+02	5.0000000D+04	1.0000000D+00	3.8300000D+05	1.0000000D+00
5	0.0	-2.8000000D-04	8.4400000D+02	5.0000000D+04	2.1800000D+01	8.3500000D+06	1.0000000D+02

LIQUID SEGMENTS IN PRIMARY LOOPS

LIQUID SEGMENT	FIRST ELEMENT	NUMBER OF ELEMENTS	JCVL ONE	JCVL TWO	FLOSSL KG/S	ZINL M	CVLMLT ONE	CVLMLT TWO
1	1	1	1	2	4.7850000D+03	-2.9960000D+00	1.0000000D+00	1.0000000D+00
2	2	3	1	2	1.2900000D+02	-2.9960000D+00	1.0000000D+00	1.0000000D+00
3	5	3	1	2	3.0300000D+02	-2.9960000D+00	1.0000000D+00	1.0000000D+00
4	8	1	2	3	1.7390000D+03	4.7560000D+00	3.0000000D+00	1.0000000D+00
5	9	6	3	1	1.7390000D+03	4.8360000D+00	1.0000000D+00	3.0000000D+00

SAS4A 0.0 OUTLET PLENUM WITH COVER GAS 2/07/84 03.22.44

GAS SEGMENTS IN PRIMARY LOOPS

SEGMENT	JCVG ONE	JCVG TWO	XLENGG M	AREASG M**2	CVGHLT ONE	CVGHLT TWO	DHSEGG M	ROUGHG M
1	2	4	1.500000D+01	3.491500D-03	1.000000D+00	1.000000D+00	6.667500D-02	0.0
2	3	4	1.500000D+01	3.491500D-03	1.000000D+00	3.000000D+00	6.667500D-02	0.0
3	4	5	5.000000D+00	6.658400D-03	1.000000D+00	1.000000D+00	9.207500D-02	0.0

PUMPS

IELPMP	IENPMP	ILRPMP
9	1	0

PUMP 1 CENTRIFUGAL PUMP

PMPINR KG-M**2	HEADR PA	PMPSPR RAD/S	PMPFLR KG/S	PMPEFR	TRKLSC J-S/KG	EPSCAV
1.200000D+03	1.126753D+06	1.168680D+02	1.932000D+03	9.000000D-01	1.000000D-01	0.0

J	APMPHD HEAD	J	APMPHD TORQUE	J	AMOTTK TORQUE	J	AMOTTK SEC
1	1.229000D+00	11	5.840000D-01	1	1.000000D+00	11	0.0
2	-1.800000D-01	12	3.390000D-02	2	0.0	12	1.000000D-02
3	1.920000D-01	13	5.530000D-01	3	0.0	13	1.000000D+03
4	-2.800000D-01	14	-1.520000D-01	4	3.000000D+00	14	2.400000D+01
5	3.900000D-02	15	-1.900000D-02	5	4.000000D+00	15	2.800000D+01
6	3.000000D+00	16	8.000000D-03	6	5.000000D+00	16	3.200000D+01
7	-2.200000D-01	17	8.000000D-03	7	6.000000D+00	17	4.000000D+02
8	1.144000D-01	18	0.0	8	8.000000D+00	18	4.010000D+02
9	8.050000D-02	19	0.0	9	1.000000D+01	19	1.000000D+04
10	6.020300D-02	20	3.240000D-02	10	1.200000D+01	20	1.900000D+01

TEMPERATURE GROUPS

TEMP. GROUP	NUMBER NODES	FIRST ELEMENT	LAST ELEMENT
1	4	2	2
2	7	3	3
3	4	4	4
4	4	5	5
5	7	6	6
6	4	7	7
7	20	8	8
8	16	9	10
9	41	11	11
10	30	12	14

BYPASS CHANNELS

CHANNEL	NTLNBY	IDKTYP	IELBYP
1	7	1	3
2	7	1	6

SAS4A 0.0 OUTLET PLENUM WITH COVER GAS 2/07/84 03.22.44

CHANNEL	XKALBY W/M-K	XKAUDY W/M-K	XKBLBY W/M-K	XKBUBY W/M-K	XKDBY W/M-K	DABY M	DBBY M
1	1.962000D+01	1.962000D+01	1.962000D+01	1.962000D+01	1.962000D+01	6.130000D-04	6.130000D-04
2	1.962000D+01	1.962000D+01	1.962000D+01	1.962000D+01	1.962000D+01	6.130000D-04	6.130000D-04

CHANNEL	DDBY M	RCALBY J/M**3-K	RCAUDY J/M**3-K	RCBLBY J/M**3-K	RCBUBY J/M**3-K	RCDBY J/M**3-K
1	6.130000D-04	6.900000D+06	6.900000D+06	6.900000D+06	6.900000D+06	6.900000D+06
2	6.130000D-04	6.900000D+06	6.900000D+06	6.900000D+06	6.900000D+06	6.900000D+06

CHANNEL	PRFRA1	PRFRA2	PERABY M	PERDBY M
1	0.0	0.0	1.830000D-02	1.830000D-02
2	0.0	0.0	1.830000D-02	1.830000D-02

CHANNEL 1

NODE	PSHPBY
1	1.000000D+00
2	1.000000D+00
3	1.000000D+00
4	1.000000D+00
5	1.000000D+00
6	1.000000D+00
7	1.000000D+00

REGION	GAMMBY	GAMDBY
1	9.775500D-04	0.0
2	0.0	0.0
3	0.0	0.0

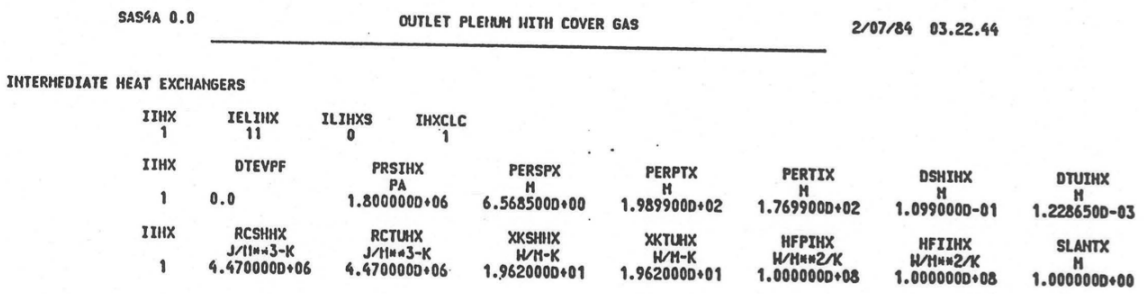
CHANNEL 2

NODE	PSHPBY
1	1.000000D+00
2	1.000000D+00
3	1.000000D+00
4	1.000000D+00
5	1.000000D+00
6	1.000000D+00
7	1.000000D+00

REGION	GAMMBY	GAMDBY
1	0.0	0.0
2	0.0	0.0
3	9.775500D-04	0.0

BOTH IHX AND DRACS: NODE, FRACTIONAL AXIAL NODE HEIGHTS

1	0.100000D+01	2	0.100000D+01	3	0.100000D+01	4	0.100000D+01	5	0.100000D+01	6	0.100000D+01
7	0.100000D+01	8	0.100000D+01	9	0.100000D+01	10	0.100000D+01	11	0.100000D+01	12	0.100000D+01
13	0.100000D+01	14	0.100000D+01	15	0.100000D+01	16	0.100000D+01	17	0.100000D+01	18	0.100000D+01
19	0.100000D+01	20	0.100000D+01	21	0.100000D+01	22	0.100000D+01	23	0.100000D+01	24	0.100000D+01
25	0.100000D+01	26	0.100000D+01	27	0.100000D+01	28	0.100000D+01	29	0.100000D+01	30	0.100000D+01
31	0.100000D+01	32	0.100000D+01	33	0.100000D+01	34	0.100000D+01	35	0.100000D+01	36	0.100000D+01
37	0.100000D+01	38	0.100000D+01	39	0.100000D+01	40	0.100000D+01	41	0.100000D+01		



SAS4A 0.0 OUTLET PLENUM WITH COVER GAS 2/07/84 03.22.44

INTERMEDIATE HEAT EXCHANGERS

IIHX	IELIHX	ILIHXS	IHXCLC
1	11	0	1

IIHX	DTEVPF	PRSIHX PA	PERSPX M	PERPTX M	PERTIX M	DSHIHX M	DTUIHX M
1	0.0	1.800000D+06	6.568500D+00	1.989900D+02	1.769900D+02	1.099000D-01	1.228650D-03

IIHX	RCSHHX J/M**3-K	RCTUHX J/M**3-K	XKSHIX W/M-K	XKTUHX W/M-K	HFPIHX W/M**2/K	HFIIHX W/M**2/K	SLANTX M
1	4.470000D+06	4.470000D+06	1.962000D+01	1.962000D+01	1.000000D+08	1.000000D+08	1.000000D+00

5.13.4 Plotting PRIMAR-4 Data

The PRIMAR4.dat binary output file is created by the PRIMAR-4 module when the input parameter NBINOT > 0. Data written to PRIMAR4.dat can be extracted with the PRIMAR4toXML utility which converts PRIMAR4.dat from binary to an XML spreadsheet. The XML spreadsheet can be imported into Microsoft Excel for convenient post-processing.

Once imported into Microsoft Excel, the XML version of PRIMAR4.dat will contain one row for each time step that was printed. The first few columns provide information related to the time, and the remaining columns provide the results of the variables selected in the IBINTOT array. They are in the same order that they are input, e.g. from K = 1 to NBINOT. In particular,

- Column A: ISTEP time step number
- Column B: IPRSTP printed time step number
- Column C: Time (sec)
- Column D: Data associated with IBINOT(1)
- Column E: Data associated with IBINOT(2)
- And so on up to IBINOT(NBINOT)

Unlike other binary output files, only user-selected data appears on PRIMAR4.dat. That is, if you do not request a particular data item (e.g. liquid segment flow rate) to be written, then it will not appear in PRIMAR4.dat. The user selects the data items in the INPMR4 input block (Block 3) using inputs **NBINOT**, **IBINST**, **IBINOT**, and **INULLT**. The input specifications are shown in Table 2.7.1. The appropriate values for IBNTYP and INUM for each available variable are provided in Table 2.7.2.

SAS4A 0.0 OUTLET PLENUM WITH COVER GAS 2/07/84 03.22.44

********** STEADY-STATE PRIMAR-4 PRINTOUT **********

COMPRESSIBLE VOLUMES

C.V.	LIQUID PRESSURE	GAS PRESSURE	LIQUID TEMP.	GAS TEMP.	LIQUID MASS	GAS MASS	GAS VOLUME	TOTAL VOLUME	LIQ.-GAS INTERFACE	REF. PRESSURE	REF.TEMP. OR HEIGHT	TYPE
1	1028019.0	0.0	599.1500	660.9000	75350.0	0.0	0.0	86.40000	-1.60000	1028019.0	599.15000	1
2	155300.0	100700.0	794.7701	794.7701	57907.6	36.53147	60.00000	130.00000	9.74830	0.0	0.0	7
3	136791.9	100700.0	794.7701	794.7701	8603.4	3.40960	5.60000	16.00000	9.29679	0.0	0.0	9
4	0.0	109700.0	808.2000	808.2000	0.0	0.59874	1.00000	1.00000	14.36000	0.0	808.20000	11
5	0.0	100700.0	808.2000	808.2000	0.0	59.87404	100.00000	100.00000	14.36000	0.0	808.20000	11

COMPRESSIBLE VOLUMES (CONTINUED)

ICV	ALPHAP	ALPHAT	ZCVL	AREAIN	BTAPNA	BTATNA	RHONAR	NWALL	AWALL	CMWALL
1	0.11000D-08	0.0	-0.16000D+01	0.0	0.21300D-09	-0.28000D-03	0.84400D+03	0.50000D+05	0.19800D+02	0.75900D+07
2	0.0	0.0	0.30000D+01	0.27000D+02	0.21300D-09	-0.28000D-03	0.84400D+03	0.10000D+04	0.39300D+02	0.13000D+09
3	0.0	0.0	0.48360D+01	0.30700D+01	0.21300D-09	-0.28000D-03	0.84400D+03	0.50000D+05	0.64000D+01	0.24500D+07
4	0.0	0.0	0.14360D+02	0.10000D+01	0.0	-0.28000D-03	0.84400D+03	0.50000D+05	0.10000D+01	0.38300D+06
5	0.0	0.0	0.14360D+02	0.10000D+02	0.0	-0.28000D-03	0.84400D+03	0.50000D+05	0.21800D+02	0.83500D+07

COMPRESSIBLE VOLUMES (CONTINUED)

ICV	VOLGSO	ZLQREF
1	0.0	0.0
2	6.000000D+01	7.155703D+00
3	5.600000D+00	5.909168D+00
4	1.000000D+00	0.0
5	1.000000D+02	0.0

LIQUID SEGMENTS

ISGL	FLOSSL	FLOSL2	ZINL	CVLMLT 1	CVLMLT 2	JCVL 1	JCVL 2	TSLI 1	TSLI 2
1	4.785000D+03	4.784850D+03	-2.996000D+00	1.000000D+00	1.000000D+00	1	2	0.0	0.0
2	1.290000D+02	1.289960D+02	-2.996000D+00	1.000000D+00	1.000000D+00	1	2	5.991500D+02	6.079693D+02
3	3.030000D+02	3.029905D+02	-2.996000D+00	1.000000D+00	1.000000D+00	1	2	5.991500D+02	6.029030D+02
4	1.739000D+03	1.738945D+03	4.756000D+00	3.000000D+00	1.000000D+00	2	3	7.947701D+02	7.947701D+02
5	1.739000D+03	1.738945D+03	4.836000D+00	1.000000D+00	3.000000D+00	3	1	7.947701D+02	5.991500D+02

SAS4A 0.0 OUTLET PLENUM WITH COVER GAS 2/07/84 03.22.44

A TABLE FOR LIQUID ELEMENTS

ISGL	IELL	ZOUTEL	GRAVIID	TELE 1	TELE 2	XLEHEL	AREAEL	DHELEM	ROUGHL	BENDNM	G2PRDR	WALLMC	WALLH	DPRSEL
1	1	3.000	0.0	0.0	0.0	5.996	1.0000	1.0000	0.0000	0.0	0.0	109550.0	183530.0	0.0
2	2	0.145	26791.6	599.150	599.150	3.141	0.0100	0.1000	0.0000	1.000	6.7	109550.0	183530.0	741010.2
	3	1.100	8136.4	599.150	607.969	0.955	0.0100	0.1000	0.0000	0.0	0.0	109550.0	183530.0	15912.8
	4	2.700	13616.1	607.969	607.969	1.600	0.0100	0.1000	0.0000	1.000	0.0	109550.0	183530.0	76732.0
3	5	0.145	26791.6	599.150	599.150	3.141	0.0100	0.1000	0.0000	1.000	-0.5	109550.0	183530.0	324713.6
	6	1.100	8141.8	599.150	602.903	0.955	0.0100	0.1000	0.0000	0.0	0.0	109550.0	183530.0	87457.1
	7	2.700	13634.1	602.903	602.903	1.600	0.0100	0.1000	0.0000	1.000	0.0	109550.0	183530.0	421460.9
4	8	4.836	647.3	794.770	794.770	33.528	0.6207	0.8890	0.0000	1.000	0.0	165490.0	279290.0	3653.1
5	9	4.836	0.0	794.770	794.770	1.000	0.2680	0.5842	0.0000	0.0	0.0	109550.0	183530.0	0.0
	10	9.326	36326.7	794.770	794.770	24.384	0.2680	0.5842	0.0000	10.900	0.0	109550.0	183530.0	114065.6
	11	1.794	-62567.0	794.770	599.150	8.820	1.9360	0.0377	0.0000	0.0	0.0	109550.0	183530.0	2409.2
	12	-0.900	-22981.4	599.150	599.150	22.360	0.2680	0.5842	0.0000	6.800	0.0	109550.0	183530.0	71262.9
	13	-0.900	0.0	599.150	599.150	1.000	0.2680	0.5842	0.0000	1.000	0.0	109550.0	183530.0	9327.7
	14	-2.990	-17822.7	599.150	599.150	22.360	0.2680	0.5842	0.0000	6.800	0.0	109550.0	183530.0	71262.9

RESTART NUMBER 1 HAS BEEN WRITTEN ON LOGICAL UNIT 17 ON MAIN TIME STEP 0

SAS4A 0.0 OUTLET PLENUM WITH COVER GAS 2/07/84 03.22.44

***** PRIMAR-4 RESULTS *****

MAIN TIME STEP 50 PRIMAR TIME STEP 50 TIME 4.749631D+01 SECONDS

COMPRESSIBLE VOLUMES, LIQUID

LIQUID-GAS INTERFACE

COMP. VOL.	PRESSURE PA	TEMPERATURE K	MASS KG	SUM3	SUM4	MASS CHANGE RATE	ZINTER M	WALL TEMP. K
1	1.974937D+05	5.991500D+02	7.526784D+04	3.112304D+05	-5.195082D+02	-5.501158D-02	-1.600000D+00	5.991500D+02
2	1.452315D+05	7.108258D+02	5.880635D+04	1.693435D+05	-2.816176D+02	1.195305D+01	9.728352D+00	7.940630D+02
3	1.302360D+05	7.947378D+02	8.331217D+03	8.044368D+04	-1.052175D+02	-3.966013D+00	9.189582D+00	7.947506D+02
4	0.0	8.082000D+02	0.0	0.0	0.0	0.0	1.436000D+01	8.082000D+02
5	0.0	8.082000D+02	0.0	0.0	0.0	0.0	1.436000D+01	8.082000D+02

COMPRESSIBLE VOLUMES, GAS

COMP. VOL.	PRESSURE PA	TEMPERATURE K	MASS KG	XISUM	THSUM
1	0.0	6.609000D+02	0.0	3.653364D+01	7.121351D+02
2	8.943214D+04	7.121311D+02	3.653364D+01	3.409909D+00	7.943106D+02
3	9.505105D+04	7.942894D+02	3.409909D+00	5.958097D-01	8.036937D+02
4	1.002995D+05	8.091348D+02	5.956336D-01	5.987408D+01	8.082010D+02
5	1.007003D+05	8.082022D+02	5.987405D+01	0.0	0.0

LIQUID SEGMENTS

LIQ. SEG.	FLOW RATE KG/S	TEMP. IN K	TEMP. OUT K
1	0.0	5.991500D+02	7.108258D+02
2	1.618086D+01	5.991500D+02	6.013777D+02
3	3.810424D+01	5.991500D+02	6.001274D+02
4	1.869165D+02	7.108258D+02	7.946086D+02
5	1.909241D+02	7.947378D+02	5.991500D+02

GAS SEGMENTS

GAS SEG.	FLOW RATE KG/S
1	-6.964682D-05
2	-3.456854D-05
3	-2.971229D-05

PUMPS

PUMP	HEAD	FLOW	SPEED	PUMP TORQUE	MOTOR TORQUE	NORM. SPEED	NORM. FLOW	NORM. HEAD
1	14590.78	190.9241	12.9891	0.0119484	0.0	0.1111431	0.0988220	0.0129494

PIPE-TYPE TEMPERATURES, K TEMPERATURE GROUP 1 FIRST ELEMENT 2 LAST ELEMENT 2

J	COOLANT	WALL	J	COOLANT	WALL	J	COOLANT	WALL	J	COOLANT	WALL	J	COOLANT	WALL
1	599.150	599.150	2	599.150	599.150	3	599.150	599.150	4	599.150	599.150			

SAS4A 0.0 OUTLET PLENUM WITH COVER GAS 2/07/84 03.22.44

BYPASS TEMPERATURES, K TEMPERATURE GROUP 2 ELEMENT NUMBER 3

J	OUTER REFLECTOR	INNER REFLECTOR	COOLANT	DUCT WALL	J	OUTER REFLECTOR	INNER REFLECTOR	COOLANT	DUCT WALL	J	OUTER REFLECTOR	INNER REFLECTOR	COOLANT	DUCT WALL
1	843.715	844.028	599.150	599.322	2	844.075	844.389	599.497	599.667	3	844.435	844.750	599.843	600.011
4	844.794	845.110	600.189	600.354	5	845.153	845.470	600.535	600.698	6	845.511	845.829	600.880	601.040
7	845.869	846.183	601.225	601.382										
8			601.569											

PIPE-TYPE TEMPERATURES, K TEMPERATURE GROUP 3 FIRST ELEMENT 4 LAST ELEMENT 4

J	COOLANT	WALL	J	COOLANT	WALL	J	COOLANT	WALL	J	COOLANT	WALL	J	COOLANT	WALL
1	601.536	601.093	2	601.473	601.109	3	601.419	601.136	4	601.378	601.173			

PIPE-TYPE TEMPERATURES, K TEMPERATURE GROUP 4 FIRST ELEMENT 5 LAST ELEMENT 5

J	COOLANT	WALL	J	COOLANT	WALL	J	COOLANT	WALL	J	COOLANT	WALL	J	COOLANT	WALL
1	599.150	599.150	2	599.150	599.150	3	599.150	599.150	4	599.150	599.150			

BYPASS TEMPERATURES, K TEMPERATURE GROUP 5 ELEMENT NUMBER 6

J	OUTER REFLECTOR	INNER REFLECTOR	COOLANT	DUCT WALL	J	OUTER REFLECTOR	INNER REFLECTOR	COOLANT	DUCT WALL	J	OUTER REFLECTOR	INNER REFLECTOR	COOLANT	DUCT WALL
1	599.223	599.223	599.150	754.308	2	599.370	599.370	599.298	754.459	3	599.516	599.517	599.446	754.611
4	599.662	599.654	599.594	754.761	5	599.809	599.810	599.742	754.912	6	599.955	599.957	599.890	755.063
7	600.101	600.103	600.037	755.214										
8			600.185											

PIPE-TYPE TEMPERATURES, K TEMPERATURE GROUP 6 FIRST ELEMENT 7 LAST ELEMENT 7

J	COOLANT	WALL	J	COOLANT	WALL	J	COOLANT	WALL	J	COOLANT	WALL	J	COOLANT	WALL
1	600.181	599.993	2	600.167	599.983	3	600.147	599.970	4	600.127	599.958			

PIPE-TYPE TEMPERATURES, K TEMPERATURE GROUP 7 FIRST ELEMENT 8 LAST ELEMENT 8

J	COOLANT	WALL	J	COOLANT	WALL	J	COOLANT	WALL	J	COOLANT	WALL	J	COOLANT	WALL
1	712.124	737.399	2	715.727	741.221	3	720.529	746.439	4	725.703	751.686	5	731.209	756.892
6	736.961	761.983	7	742.937	766.884	8	748.994	771.520	9	755.052	775.828	10	761.160	779.737
11	766.890	783.193	12	772.248	786.182	13	777.457	788.675	14	782.079	790.640	15	785.722	792.124
16	788.645	793.214	17	791.050	793.963	18	792.908	794.414	19	794.008	794.637	20	794.609	794.719

PIPE-TYPE TEMPERATURES, K TEMPERATURE GROUP 8 FIRST ELEMENT 9 LAST ELEMENT 10

J	COOLANT	WALL	J	COOLANT	WALL	J	COOLANT	WALL	J	COOLANT	WALL	J	COOLANT	WALL
1	794.742	794.755	2	794.747	794.758	3	794.750	794.760	4	794.753	794.761	5	794.756	794.763
6	794.758	794.764	7	794.760	794.765	8	794.762	794.766	9	794.763	794.767	10	794.765	794.768
11	794.766	794.768	12	794.767	794.769	13	794.767	794.769	14	794.768	794.769	15	794.768	794.770
16	794.769	794.770												

SAS4A 0.0 OUTLET PLENUM WITH COVER GAS 2/07/84 03.22.44

IHX TEMPERATURES, K TEMPERATURE GROUP 9 PRIMARY SIDE ELEMENT NUMBER 11 INTERMEDIATE SIDE ELEMENT NUMBER 0

J	SHELL	SHELL COOLANT	TUBE	TUBE COOLANT	J	SHELL	SHELL COOLANT	TUBE	TUBE COOLANT	J	SHELL	SHELL COOLANT	TUBE	TUBE COOLANT
1	0.0	794.769	599.150	599.149	2	0.0	599.149	599.150	599.149					
3		599.149		599.150										

PIPE-TYPE TEMPERATURES, K TEMPERATURE GROUP 10 FIRST ELEMENT 12 LAST ELEMENT 14

J	COOLANT	WALL	J	COOLANT	WALL	J	COOLANT	WALL	J	COOLANT	WALL	J	COOLANT	WALL
1	599.149	599.150	2	599.149	599.150	3	599.149	599.150	4	599.150	599.150	5	599.150	599.150
6	599.150	599.150	7	599.150	599.150	8	599.150	599.150	9	599.150	599.150	10	599.150	599.150
11	599.150	599.150	12	599.150	599.150	13	599.150	599.150	14	599.150	599.150	15	599.150	599.150
16	599.150	599.150	17	599.150	599.150	18	599.150	599.150	19	599.150	599.150	20	599.150	599.150
21	599.150	599.150	22	599.150	599.150	23	599.150	599.150	24	599.150	599.150	25	599.150	599.150
26	599.150	599.150	27	599.150	599.150	28	599.150	599.150	29	599.150	599.150	30	599.150	599.150

LIQUID ELEMENT PRESSURE DROPS, PA

ELEM.	PRESSURE DROP	GRAVITY HEAD	ELEM.	PRESSURE DROP	GRAVITY HEAD	ELEM.	PRESSURE DROP	GRAVITY HEAD	ELEM.	PRESSURE DROP	GRAVITY HEAD	ELEM.	PRESSURE DROP	GRAVITY HEAD
1	0.0	0.0	2	12148.7	26791.6	3	268.8	8143.2	4	1295.9	13639.2	5	5472.7	26791.6
6	1457.3	8144.7	7	7022.2	13643.9	8	47.6	653.6	9	-15148.8	0.0	10	1492.3	36326.8
11	40.4	-62567.0	12	946.1	-22981.4	13	123.8	0.0	14	946.1	-17822.7			

5.14 References

5-1. Koenig, J. F., Unpublished information, Argonne National Laboratory, 1982.

5-2. Moody, L. F., *Mechanical Engineering,* vol. 69, p. 1005, 1947.

5-3. Shapiro, A. H., *The Dynamics and Thermodynamics of Compressible Fluid Flow*, Vol. 1, p. 182, The Ronald Press Company, New York, 1953.

5-4. Rohsenow, W. M. and J. P. Hartnett, Eds., *Handbook of Heat Transfer*, p. 7-33, McGraw-Hill Book Company, New York, 1973.

5-5. Mohr, D. and E. E. Feldman, "A Dynamic Simulation of the EBR-II Plant During National Convection with the NATDEMO Code,"*Decay Heat Removal and Natural Convection in Fast Breeder Reactors*, A. K,. Agrawal and J. G. Guppy editors, Hemisphere Publishing Co., pp 207-223, 1981.

5-6. Warren M. Rohsenow and Harry Choi, *Heat, Mass, and Momentum Transfer*, Prentice-Hall, Inc., Englewood Cliffs, New Jersey, 1961.

5-7. J. J. Lorenz and P. A Howard, "Entrainment by a Jet at a Density Interface in a Thermally Stratified Vessel," *Transactions of the ASME*, vol. 101, p. 538, 1979.

5-8. J. W. Yang, "Penetration of a Turbulent Jet with Negative Buoyancy into the Upper Plenum of an LMFBR," *Nuclear Engineering and Design,* vol. 40, p. 297, 1977.

5-9."ZeroMQ," ZeroMQ. [Online]. Available: https://zeromq.org/. [Accessed: 16-Apr-2020].

5-10. O. J. Foust, "Sodium-NaK Engineering Handbook, Volume 1: Sodium Chemistry and Physical Properties," *Gordon and Breach*, 1972.

5-11. H. Ota et al., "Development of 160 m3/min Large Capacity Sodium-Immersed Self-Cooled Electromagnetic Pump," *Journal of Nuclear Science and Technology*, vol. 41, 2004.

5.15 Nomenclature

Symbol	Description	Units
A	Coolant flow area	m^2
A_{CP}, A_{CI}	Primary and intermediate coolant flow areas in the intermediate heat exchanger	m^2
A_k	Coolant flow area of element k in a liquid segment	m^2
A_w	Wall area of a compressible volume	m^2
A_{inter}	Area of liquid-gas interface	m^2
$a_0(i)$		1/m
$a_1(i)$	Coefficients for liquid segment i.	1/m-s
$a_2(i)$		1/m-s
$a_3(i)$		1/m
$a_1(j), ...,$ $a_4(j)$	Coefficients in the simultaneous equations for the temperature changes at node j in the intermediate heat exchanger	J/m^2-K
$\Delta a_0(k)$		1/m
$\Delta a_1(k)$	Contributions from element k in liquid segment i to the	1/m-s
$\Delta a_2(k)$	coefficients $a_0\,(i)$, …, $a_3(i)$	1/m-s
$\Delta a_3(k)$		1/m
$b_0(j)$	Coefficients in the pressure change expression for	Pa

continues on next page

Table 5.15.1 – continued from previous page

Symbol	Description	Units
$b_1(j)$	Compressible volume j	kg/m-s
$b_2(j)$		kg/m-s-K
C_1, C_2, C_3	Coefficients in electrical resisitivity correlation	Ω/m, Ω/m-K, Ω/m-K^2
C	Electrical resisitivity proportionality constant	–
$c_{\rm c}$, $c_{\rm l}$	Specific heat of coolant or liquid	J/kg-K
$c_{\rm w}$	Specific heat of compressible volume or pipe wall	J/kg-K
$\overline{c}_{\rm P}$, $\overline{c}_{\rm I}$	Spatially averaged primary and intermediate coolant heat capacities in the intermediate heat exchanger	J/kg-K
$\overline{c}$	Spatially averaged coolant heat capacity in the bypass	J/kg-K
C_1, C_2, C_3	User-supplied correlation coefficients in film heat-transfer calculation	–
$c_1(j), ...,$ $c_6(j)$	Coefficients in the simultaneous equations for the temperature changes at node j in the intermediate heat exchanger	J/m-K
$C_0(L, ic)$	Coefficients in the expression for core channel	kg/s^2
$C_1(L, ic)$	flow estimate in channel ic at end L	m
$C_2(L, ic)$	L = 1 for inlet, 2 for outlet	m
$C_3(L, ic)$		1/kg
$c_{\rm ij}$	Matrix in cover-gas pressure change equation	—
$c(I, J)$	Matrix in compressible volume pressure change equation for liquid flow	kg
$\Delta c(I, J)$	Contribution to the matrix $c(i, j)$	kg
D_1, D_2, D_3	Right side of simultaneous equation solution in the cover gas, intermediate heat exchanger, and bypass channel treatments	W/m
D, $D_{\rm h}$	Hydraulic diameters	m
DRACS	Direct reactor auxiliary cooling system	
$d_{\rm A}$, $d_{\rm B}$, $d_{\rm D}$	Reflector thicknesses in the bypass channel	m
$d_{\rm SH}$, $d_{\rm TU}$	Shell and tube thicknesses in the intermediate heat exchanger	m
d_1	Coefficients in coolant-wall temperature	J/m-K
d_2	Calculations	J/m^2-K
d_5		m-K/J
$d_{\rm j}$	Right side of cover-gas pressure change matrix equation	Pa
$d(j)$	Right side of liquid flow pressure change matrix equation	kg^2/m-s^2
$\Delta d(j)$	Contribution to $d(j)$	kg^2/m-s^2
$E_{\rm src}$	Heat source	W/m^3
$e_1(j), ...,$ $e_{10}(j)$	Coefficients in the simultaneous equations for the temperature changes at node j in the intermediate heat exchanger	
$e_0(L, ic)$	Coefficients in the expression for core channel	kg/s
$e_1(L, ic)$	flow estimate in channel ic at end L	m-s
$e_2(L, ic)$		m-s
$F_{\rm ij}$	Cover-gas flow rate from compressible volume i to compressible volume j	kg/s
$F_{\rm 0ij}$		kg/s
$F_{\rm 1ij}$	Coefficients in the expression for $F_{\rm ij}$	m-s
$F_{\rm 2ij}$		m-s
F_Q	Fraction of pump heat to be modeled	–
f	Moody friction factor	–
	power source frequency	Hz
$f_{\rm r}$	Fraction of a coolant node at pipe inlet	–

continues on next page

Table 5.15.1 – continued from previous page

Symbol	Description	Units
$f_1(j), ..., f_8(j)$	Coefficients in the simultaneous equations for the temperature changes at node j in the intermediate heat exchanger	
f_1, f_2	Fractions of reactor power	–
G	Specific flow rate	kg/m^2-s
GH	Gravity head	m
G_2	Pressure drop parameter	–
g	Acceleration due to gravity	m/s^2
H	Head	m
$H_{\rm r}$	Pump reference head	m
$H_{\rm SP}$, $H_{\rm PT}$, $H_{\rm TI}$	Overall heat-transfer coefficients for the intermediate heat exchanger	W/m^2-K
$H_{\rm AB}$, $H_{\rm BC}$, $H_{\rm CD}$	Overall heat-transfer coefficients between regions in the bypass channel	W/m^2-K
$H_{\rm snk}A_{\rm snk}$	Heat transfer coefficient times area	W/K
$(hA)_{\rm snk}$	Heat transfer coefficient times area	W/K
$H_{\rm D}$	Steam drum total enthalpy	J
$h_{\rm D}$	Specific enthalpy	J/kg
$h_{\rm f}$	Liquid specific enthalpy	J/kg
$h_{\rm g}$	Vapor specific enthalpy	J/kg
$h_{\rm c}$, $h_{\rm CP}$, $h_{\rm CI}$	Coolant film coefficients	W/m^2-K
$h_{\rm w}$, $h_{\rm wc}$	Wall and wall-to-coolant heat-transfer coefficients	W/m^2-K
$h_{\rm FS}$, $h_{\rm fFT}$	Fouling factors in the intermediate heat exchanger	W/m^2-K
I	Line-to-line total pump current	A
I_J	Line-to-line pump current through R_J	A
I_D	Line-to-line pump current through R_D	A
IHX	Intermediate heat exchanger	
$K_{\rm cold}$	A cold leg pressure loss coefficient	s^2/m
$k_{\rm c}$	Coolant thermal conductivity	W/m-K
$k_{\rm SH}$, $k_{\rm TU}$	Shell and tube thermal conductivities in the intermediate heat exchanger	W/m-K
L, $L_{\rm i}$	Length	m
$\Delta L(N)$	Length-integrated volumetric heat source	W/m^2
$L_{\rm k}$	Length of element k in a liquid flow segment	m
$L_{\rm n}$	Length of a node at pipe inlet	m
$L_{\rm B}/D_{\rm B}$	Length-to-diameter ratio per bend in an element	–
L_1, L_2	Inductances in equivalent circuit pump models	H
M	Mach number for cover gas flow	–
$M_{\rm D}$	Steam drum total mass	kg
$M_{\rm w}$	Pipe wall mass per unit length	kg/m
$\delta_{\rm m}$	Change in liquid mass	kg
$\delta m_{\rm i}$	Change in cover gas mass in compressible volume i during a sub-interval of time	kg
Δm, $\Delta m_{\rm l}$	Change in liquid mass	kg
$\Delta m_{\rm ji}$	Mass of cover gas flowing from compressible volume j to i during a sub-interval of time	kg
$m(i)$	Cover-gas mass in compressible volume i	kg
m	Number of evaporators in parallel	–
$m_{\rm g}$	Gas mass in a compressible volume	kg
$m_{\rm r}$	Liquid mass in a compressible volume at a reference pressure	kg

continues on next page

Table 5.15.1 – continued from previous page

Symbol	Description	Units
Δm	Change in liquid mass in a compressible volume during a time sub-interval	kg
N_{B}	Number of bends in an element	–
P, P_{er}, P_{A}, P_{D}, P_{SP}, P_{TI}	Perimeters	m
P_{s}	Power shape for vertical section	–
p, p_{l}	Liquid pressure in a compressible volume	Pa
p_{d}, p_{d1}, p_{d2}	Pump head coefficients	
p_{r}	Reference liquid pressure in a compressible volume	Pa
Δp	Change in liquid pressure in a compressible volume during time sub-interval	Pa
$p(i)$, p_{g}	Cover-gas pressure in compressible volume i	Pa
P_1, P_2	Power shape normalization for Regions 1 and 2	
$p(\mathrm{JIN})$, $p(\mathrm{JX})$	Inlet and outlet plenum pressures in core flow estimate	Pa
Δp_{i}	Change in cover-gas pressure in compressible volume i during a sub-interval of time	Pa
$p_{\mathrm{in}}(i)$, $p_{\mathrm{out}}(i)$	Pressures at the inlet and outlet of liquid segment i	Pa
δp	Change in liquid pressure	Pa
$\Delta p_{\mathrm{fr}}(i)$	Frictional pressure loss in liquid segment i	Pa
$\Delta p_{\mathrm{w2}}(i)$	Pressure loss proportional to the square of the flow in liquid segment i	Pa
$\Delta p_{\mathrm{v}}(i)$	Valve pressure loss in liquid segment i	Pa
$\Delta p_{\mathrm{gr}}(i)$	Gravity head in liquid segment i	Pa
$\Delta p_{\mathrm{p}}(i)$	Pump head in liquid segment i	Pa
$P_{\mathrm{TOT}} Q_{\mathrm{MULT}}$	Reactor power for a time step	W
$P_{\mathrm{s}}(j)$	Power shape by nodes	–
$P1$, $P3$	Power shape normalization factors	–
Q_{A}, Q_{B}, Q_{D}	Neutron and decay heating sources in reflectors	W/m^3
QMULT	Power multiplication factor	–
q'_{c}	Linear heat rate in liquid element fluid	W/m
q'_{w}	Linear heat rate in liquid element wall	W/m
q'_{wa}	Linear heat rate in liquid element wall a	W/m
q'_{wb}	Linear heat rate in liquid element wall b	W/m
q, Q	Heat flow from compressible volume walls	W/m^2
	Region length-integrated sodium heat source w	W/m^2
	Length-integrated volumetric heat source	W/m^2
Q	Heat removal rate	–
	Volumetric flow rate	m^3/s
R	Universal gas constant	J/kg-K
R_1, R_D, R_J, R_M	Circuit resistances in equivalent circuit models	Ω
Re	Reynolds number	–
Ri	Richardson number	–
S	Slant height ratio of the tube-side in the intermediate heat exchanger	–
s	Pump speed	1/s
	Pump slip	–
T	Pump torque	N-m
T, T_{g}	Cover-gas temperature	K
T, T_{l}	Liquid temperature	K
T_{c}, T_{w}	Coolant and wall temperatures	K

continues on next page

Table 5.15.1 – continued from previous page

Symbol	Description	Units
T_{f}	Friction torque loss	N-m
T_{m}, T_{p}	Motor torque, pump torque	N-m
T_{SH}, T_{TU}	Shell and tube node temperatures in the intermediate heat exchanger	K
$\overline{T}_{\mathrm{CP}}$, $\overline{T}_{\mathrm{CI}}$	Spatially averaged coolant temperatures in the intermediate heat exchange	K
$T(i)$	Cover-gas temperature in compressible volume i	K
$T_{\mathrm{in}}(j)$, $T_{\mathrm{out}}(j)$	Temperature of the incoming and outgoing fluid at compressible volume j	K
T_{ji}	Temperature of the cover-gas flowing from compressible volume j to i	K
T_{B}	Temperature of the flow from a bypass channel to the outlet plenum	K
T_{C}	Temperature of the flow from a core channel to the outlet plenum	K
T_{snk}	Temperature of the heat sink	K
$T_{\mathrm{Na,i}}$	Sodium temperature in the i-th region	K
T_{out}	Outlet temperature in outlet plenum	K
$T_{\mathrm{out}}(t)$	Steam generator outlet temperature	K
δT	Change in liquid temperature	K
ΔT	Liquid temperature change during a time sub-interval	K
ΔT_{w}	Wall temperature change during a time sub-interval	K
ΔT_{SH}, ΔT_{TU}	Shell and tube node temperature changes during a time sub-interval	K
ΔT_{CP}, ΔT_{CI}	Coolant node temperature changes during a time a sub-interval	K
Δt	Time-step size	s
Δt_{s}, δt	Sub-interval of a time step	s
U	Overall heat transfer coefficient	W/m^2-K
u	Flow velocity	m/s
$V(i)$, V_{g}	Cover-gas volume in compressible volume i	m^3
V_{D}	Steam drum volume	m^3
v_{D}	Steam drum specific volume	m^3/kg
v_{j}	Jet velocity at hot-cold interface	m/s
v	Liquid velocity	m/s
V, V_{l}	Liquid volume in a compressible volume	m^3
V_{r}	Liquid volume in a compressible volume at a reference pressure	m^3
V_s	Line-to-line voltage of pump power source	V
ΔV_{g}	Change in cover gas volume	m^3
ΔV_{l}	Change in liquid volume	m^3
w	Pump flow rate	kg/s
$w(i)$	Liquid flow rate in segment i	kg/s
$\overline{w}(i)$	Time-averaged liquid flow rate in segment i	kg/s
$\overline{w}_{\mathrm{in}}(i)$, $\overline{w}_{\mathrm{out}}(i)$	Time-averaged liquid flow rate into and out from compressible volume j	kg/s
$\overline{w}_{\mathrm{B}}$	Average flow rate from a bypass channel into the outlet plenum	kg/s
$\overline{w}_{\mathrm{C}}$	Average flow rate from a core channel into the outlet plenum	kg/s
$w_{\mathrm{c}}(L, ic)$	Core flow rate in channel ic at end L at the beginning of the timestep	kg/s
$\Delta w(i)$	Change in the liquid flow rate in segment i during a time sub-interval	kg/s
X_1, X_2	Reactances in equivalent circuit pump models	Ω
z, z_{r}, z_{ref}	Reference elevation	m
z_{i}, z_{inter}	Liquid-gas interface elevation	m
z_{in}, z_{out}	Height of element inlet and outlet	m
z_{PLENL}	Lower plenum elevation	m
z_{PLENU}	Upper plenum elevation	m

continues on next page

Table 5.15.1 – continued from previous page

Symbol	Description	Units
z_{IHX}	Reference height of the thermal center of the intermediate heat exchanger	m
$\Delta z, \Delta z(j)$	Height of the j-th node	m
α_P	Volume pressure expansion coefficient	1/Pa
α_T	Volume temperature expansion coefficient	1/K
$\alpha_1, ..., \alpha_4$	Tri-diagonal matrix coefficients in the cover gas, intermediate heat exchanger, and bypass channel treatments	W/m-K
β_P	Liquid sodium compressibility	1/Pa
β_T	Liquid sodium thermal expansion coefficient	1/K
$\beta_1, ..., \beta_4$	Tri-diagonal matrix coefficients in the cover gas, intermediate heat exchanger , and bypass channel treatments	w/m-K
γ	Ratio of specific heat at constant pressure to that at constant volume	–
γ_{N1}, γ_{N3}	Fraction of neutron power	–
γ_{D1}, γ_{D3}	Fraction of decay power	–
$\gamma(i)$	Factor in the degree of implicitness in liquid segment i	–
$\gamma_c(L, ic)$	Factor in the degree of implicitness in core channel ic at end L	–
δ_{ij}	Kronecker delta symbol	–
ϵ	Pump efficiency	–
$\varepsilon, \varepsilon(k)$	Pipe roughness factor	m
ε_i	Factor in equation for cover-gas flow in compressible volume i	–
ε_v	Cover-gas volume fraction	–
ξ	Ratio of shutoff head to design head	–
ρ, ρ_l	Liquid density	kg/m^3
$\overline{\rho}_P$, $\overline{\rho}_I$	Spatially averaged primary and intermediate coolant densities in the intermediate heat exchanger	kg/m^3
ρ_I, ρ_0	Fluid density at element inlet and outlet	kg/m^3
$(\rho c)_{SH}$, $(\rho c)_{TU}$	Shell and tube densities times specific heats in the intermediate heat exchanger	J/m^3-K
$(\rho c)_A$, $(\rho c)_B$, $(\rho c)_D$	Densities times specific heats in the bypass channel	J/m^3-K
ρ_r	Electrical resisitivity	Ω
τ	Cover-gas time constant	s
τ_m, τ_p	Motor torque, pump torque	N-m
$\theta_1(i)$, $\theta_2(i)$	Degree of implicitness in the time differencing in liquid segment i	–
$\theta_{2c}(L, ic)$	Degree of implicitness in the time differencing in core channel ic at end L	–
ϕ	Phase angle between voltage and current	–
Ψ	circuit total resistance	Ω

Subscripts 3 and 4 appended to quantities refer to the values of those quantities at the beginning and at the end, respectively, of a time interval.

5.16 Appendices

5.16.1 Appendix 5.1: IHX Matrix Coefficients

The coefficients in Eq. (5.4-44) for the j-th vertical section of the shell in terms of the quantities defined in Section 5.4.2.2 are as follows:

$$a_1(j) = (\rho c)_{\text{SH}} d_{\text{SH}} + \theta_{2\text{S}} \Delta t H_{\text{S}}(j) + \theta_{2\text{S}} \Delta t \frac{(hA)_{\text{snk}}}{P_{\text{S}}} \tag{5.16-1}$$

$$a_2(j) = -\frac{1}{2}\theta_{2\text{S}} \Delta t H_{\text{S}}(j) \tag{5.16-2}$$

$$a_3(j) = -\frac{1}{2}\theta_{2\text{S}} \Delta t H_{\text{S}}(j) \tag{5.16-3}$$

$$a_4(j) = -\Delta t H_{\text{S}}(j) T_{\text{SH3}}(j) + \Delta t H_{\text{S}}(j) \overline{T}_{\text{CS3}}(j) + \frac{\Delta t (hA)_{\text{snk}} [T_{\text{snk}} - T_{\text{SH3}}(j)]}{P_{\text{S}}} \tag{5.16-4}$$

$$\overline{T}_{\text{CP3}}(j) = \frac{1}{2}[T_{\text{CS3}}(j) + T_{\text{CS3}}(j+1)] \tag{5.16-5}$$

where

$\theta_{2\text{S}}$ = the degree of implicitness for the shell-side coolant channel

Δt = the time interval

The coefficients in Eq. (5.4-45) for the j-th vertical section of the shell-side coolant for normal flow (downward) are:

$$\begin{aligned} e_1(j) = {} & \frac{1}{2} A_{\text{c}} \overline{\rho}_{\text{CS}}(j) \overline{c}_{\text{CS}}(j) + \Delta t \frac{\overline{c}_{\text{CS}}(j)}{\Delta z(j)} \theta_{2\text{S}} |w_{\text{S4}}| \\ & + \Delta t P_{\text{S}} H_{\text{S}}(j) \frac{1}{2}\theta_{2\text{S}} + \Delta t S P_{\text{ST}} H_{\text{ST}}(j) \frac{1}{2}\theta_{2\text{S}} \end{aligned} \tag{5.16-6}$$

$$e_2(j) = 0 \tag{5.16-7}$$

$$e_3(j) = -\Delta t P_{\text{S}} H_{\text{S}}(j) \theta_{2\text{S}} \tag{5.16-8}$$

$$e_4(j) = 0 \tag{5.16-9}$$

$$e_5(j) = -\Delta t S P_{\text{ST}} H_{\text{ST}}(j) \theta_{2\text{S}} \tag{5.16-10}$$

$$e_6(j) = 0 \tag{5.16-11}$$

$$\begin{aligned} e_7(j) = {} & \frac{1}{2} A_{\text{CS}} \overline{\rho}_{\text{CS}}(j) \overline{c}_{\text{CS}}(j) - \Delta t \frac{\overline{c}_{\text{CS}}(j)}{\Delta z(j)} \theta_{2\text{S}} \mid w_{\text{S4}}| \\ & + \Delta t P_{\text{S}} H_{\text{S}}(j) \frac{1}{2}\theta_{2\text{S}} + \Delta t S P_{\text{ST}} H_{\text{ST}}(j) \frac{1}{2}\theta_{2\text{S}} \end{aligned} \tag{5.16-12}$$

$$\begin{aligned} e_8(j) = {} & -\Delta t \frac{\overline{c}_{\text{CS}}(j)}{\Delta z(j)} \{\theta_{1\text{S}} \mid w_{\text{S3}}| + \theta_{2\text{S}} |w_{\text{S4}}| [T_{\text{CS3}}(j) - T_{\text{CS3}}(j+1)]\} \\ & + \Delta t P_{\text{S}} H_{\text{S}}(j) \left\{ T_{\text{SH3}}(j) - \frac{1}{2}[T_{\text{CS3}}(j) + T_{\text{CS3}}(j+1)] \right\} \\ & + \Delta t S P_{\text{S}} H_{\text{ST}}(j) \left\{ T_{\text{TU3}}(j) - \frac{1}{2}[T_{\text{CS3}}(j) + T_{\text{CS3}}(j+1)] \right\} \end{aligned} \tag{5.16-13}$$

The same coefficients for reversed flow (upward) in the shell-side coolant channel are:

$$\begin{aligned} e\left(j\right) = \frac{1}{2}A_{\mathrm{CS}}\overline{\rho}_{\mathrm{CS}}\left(j-1\right)\overline{c}_{\mathrm{CS}}\left(j-1\right) + \Delta t\frac{\overline{c}_{\mathrm{CS}}\left(j-1\right)}{\Delta z\left(j-1\right)}\theta_{2\mathrm{S}}\left|w_{\mathrm{S4}}\right| \\ +\Delta t P_{\mathrm{S}}H_{\mathrm{S}}\left(j-1\right)\frac{1}{2}\theta_{2\mathrm{S}} + \Delta t S P_{\mathrm{ST}}H_{\mathrm{ST}}\left(j-1\right)\frac{1}{2}\theta_{2\mathrm{S}} \end{aligned} \tag{5.16-14}$$

$$e_2\left(j\right) = -\Delta t P_{\mathrm{S}}H_{\mathrm{S}}\left(j-1\right)\theta_{2\mathrm{S}} \tag{5.16-15}$$

$$e_3\left(j\right) = 0 \tag{5.16-16}$$

$$e_4\left(j\right) = -\Delta t S P_{\mathrm{ST}}H_{\mathrm{ST}}\left(j-1\right)\theta_{2\mathrm{S}} \tag{5.16-17}$$

$$e_5\left(j\right) = 0 \tag{5.16-18}$$

$$\begin{aligned} e_6\left(j\right) = \frac{1}{2}A_{\mathrm{CS}}\overline{\rho}_{\mathrm{CS}}\left(j-1\right)\overline{c}_{\mathrm{CS}}\left(j-1\right) - \Delta t\frac{\overline{c}_{\mathrm{CS}}\left(j-1\right)}{\Delta z\left(j-1\right)}\theta_{2\mathrm{S}}\left|w_{\mathrm{S4}}\right| \\ +\Delta t P_{\mathrm{S}}H_{\mathrm{S}}\left(j-1\right)\frac{1}{2}\theta_{2\mathrm{S}} + \Delta t S P_{\mathrm{ST}}H_{\mathrm{ST}}\left(j-1\right)\frac{1}{2}\theta_{2\mathrm{S}} \end{aligned} \tag{5.16-19}$$

$$e_7\left(j\right) = 0 \tag{5.16-20}$$

$$\begin{aligned} e_8\left(j\right) = -\Delta t\frac{\overline{c}_{\mathrm{CS}}\left(j-1\right)}{\Delta z\left(j-1\right)}\left\{\left(\theta_{1\mathrm{S}}\left|w_{\mathrm{S3}}\right| + \theta_{2\mathrm{S}}\left|w_{\mathrm{S4}}\right|\right)\left[T_{\mathrm{CS3}}\left(j\right) - T_{\mathrm{CS3}}\left(j-1\right)\right]\right\} \\ +\Delta t P_{\mathrm{S}}H_{\mathrm{S}}\left(j-1\right)\left\{T_{\mathrm{SS3}}\left(j-1\right) - \frac{1}{2}\left[T_{\mathrm{CS3}}\left(j-1\right) + T_{\mathrm{CS3}}\left(j\right)\right]\right\} \\ +\Delta t S P_{\mathrm{ST}}\left(j-1\right)\left\{T_{\mathrm{TU3}}\left(j-1\right) - \frac{1}{2}\left[T_{\mathrm{CS3}}\left(j-1\right) + T_{\mathrm{CS3}}\left(j\right)\right]\right\} \end{aligned} \tag{5.16-21}$$

The terms $e_9\left(j\right)$ and $e_{10}\left(j\right)$ have been added to Eq. (5.4-45) because they appear during the solution of the simultaneous equations. These arrays are set to zero before the solution is begun.

In addition, the boundary conditions for normal shell-side coolant channel flow are

$$e_1\left(jm\right) = 1; \quad e_{2,3,4,5,6,7}\left(jm\right) = 0; \quad e_8\left(jm\right) = \Delta T_{\mathrm{CS}}\left(jm\right) \tag{5.16-22}$$

For reversed primary channel flow, they are

$$e_1\left(1\right) = 1; \quad e_{2,3,4,5,6,7}\left(1\right) = 0; quade_8\left(1\right) = \Delta T_{\mathrm{CS}}\left(1\right) \tag{5.16-23}$$

and for both cases, they are

$$\begin{aligned} e_2\left(1\right) = 0; \quad e_4\left(1\right) = 0; \quad e_6\left(1\right) = 0 \\ e_3\left(jm\right) = 0; \quad e_5\left(jm\right) = 0; \quad e_7\left(jm\right) = 0 \end{aligned} \tag{5.16-24}$$

The coefficients in Eq. (5.4-46) for the j-th vertical section of the tube are:

$$\begin{aligned} c_1\left(j\right) = \left(\rho c\right)_{\mathrm{TU}}\frac{1}{2}\left(P_{\mathrm{ST}} + P_{\mathrm{TT}}\right)d_{\mathrm{TU}} + \Delta t\theta_{2\mathrm{S}}P_{\mathrm{ST}}H_{\mathrm{ST}}\left(j\right) \\ +\Delta t\theta_{\mathrm{ST}}P_{\mathrm{TT}}H_{\mathrm{TT}}\left(j\right) \end{aligned} \tag{5.16-25}$$

$$c_2\left(j\right) = -\frac{1}{2}\Delta t\theta_{2\mathrm{S}}P_{\mathrm{ST}}H_{\mathrm{ST}}\left(j\right) \tag{5.16-26}$$

$$c_3\left(j\right) = -\frac{1}{2}\Delta t\theta_{2\mathrm{S}}P_{\mathrm{ST}}H_{\mathrm{ST}}\left(j\right) \tag{5.16-27}$$

$$c_4\left(j\right) = -\frac{1}{2}\Delta t\theta_{2\mathrm{T}}P_{\mathrm{TT}}H_{\mathrm{TT}}\left(j\right) \tag{5.16-28}$$

$$c_5(j) = -\frac{1}{2}\Delta t\theta_{2T}P_{TT}H_{TT}(j) \tag{5.16-29}$$

$$\begin{aligned} c_6(j) = &-\Delta t\left[P_{ST}H_{ST}(j) + P_{TT}H_{TT}(j)\right]T_{TU3}(j) \\ &+\Delta t P_{ST}H_{ST}(j)\frac{1}{2}\left[T_{CS3}(j) + T_{CS3}(j+1)\right] \\ &+\Delta t P_{TT}H_{TT}(j)\frac{1}{2}\left[T_{CT3}(j) + T_{CT3}(j+1)\right] \end{aligned} \tag{5.16-30}$$

The coefficients in Eq. (5.4-47) for the j-th vertical section of the tube-side coolant for normal flow (upward) are:

$$\begin{aligned} f_1(j) = &\frac{1}{2}A_{CT}\overline{\rho}_{CT}(j-1)\,\overline{c}_{CT}(j-1) + \Delta t\frac{\overline{c}_{CT}(j-1)}{\Delta z(j-1)\,S}\left|w_{T4}\right|\theta_{2T} \\ &+\Delta t P_{TT}H_{TT}(j-1)\frac{1}{2}\theta_{2T} \end{aligned} \tag{5.16-31}$$

$$f_2(j) = -\Delta t P_{TT}H_{TT}(j-1)\,\theta_{2T} \tag{5.16-32}$$

$$f_3(j) = 0 \tag{5.16-33}$$

$$\begin{aligned} f_4(j) = &\frac{1}{2}A_{CT}\overline{\rho}_{CT}(j-1) - \Delta t\frac{\overline{c}_{CT}(j-1)}{\Delta z(j-1)\,S}\left|w_{T4}\right|\theta_{2T} \\ &+\Delta t P_{TT}H_{TT}(j-1)\frac{1}{2}\theta_{2T} \end{aligned} \tag{5.16-34}$$

$$f_5(j) = 0 \tag{5.16-35}$$

$$\begin{aligned} f_6(j) = &-\Delta t\frac{\overline{c}_{CT}(j-1)}{\Delta z(j-1)\,S}\left\{\left(\left|w_{T3}\right|\theta_{1T} + \left|w_{T4}\right|\theta_{2T}\right)\left[T_{CT3}(j) - T_{CT3}(j-1)\right]\right\} \\ &+\Delta t P_{TT}H_{TT}(j-1)\left\{T_{TU3}(j-1) - \frac{1}{2}\left[T_{CT3}(j-1) + T_{CT3}(j)\right]\right\} \end{aligned} \tag{5.16-36}$$

The same coefficients for reversed flow (downward) in the intermediate coolant channel are:

$$\begin{aligned} f_1(j) = &\frac{1}{2}A_{CT}\overline{\rho}_{CT}(j)\,\overline{c}_{CT}(j) + \Delta t\frac{\overline{c}_{I}(j)}{\Delta z(j)\,S}\left|w_{T4}\right|\theta_{2T} \\ &+\Delta t P_{TT}H_{TT}(j)\,\theta_{2T} \end{aligned} \tag{5.16-37}$$

$$f_2(j) = 0 \tag{5.16-38}$$

$$f_3(j) = -\Delta t P_{TT}H_{TT}(j)\,\theta_{2T} \tag{5.16-39}$$

$$f_4(j) = 0 \tag{5.16-40}$$

$$\begin{aligned} f_5(j) = &\frac{1}{2}A_{CT}\overline{\rho}_{CT}(j)\,\overline{c}_{CT}(j) - \Delta t\frac{\overline{c}_{CT}(j)}{\Delta z(j)\,S}\left|w_{T4}\right|\theta_{2T} \\ &+\Delta t P_{TT}H_{TT}(j)\frac{1}{2}\theta_{2T} \end{aligned} \tag{5.16-41}$$

$$\begin{aligned} f_6(j) = &-\Delta t\frac{\overline{c}_{CT}(j)}{\Delta z(j)\,S}\left\{\left(\left|w_{T3}\right|\theta_{1T} + \left|w_{T4}\right|\theta_{2T}\right)\left[T_{CT3}(j) - T_{CT3}(j+1)\right]\right\} \\ &+\Delta t P_{TT}H_{TT}(j)\left\{T_{TU3}(j) - \frac{1}{2}\left[T_{CT3}(j) + T_{CT3}(j+1)\right]\right\} \end{aligned} \tag{5.16-42}$$

The terms for $f_7(j)$ and $f_8(j)$ have been added to Eq. (5.4-47) because they appear during the solution of the simultaneous equations. These arrays are also set to zero before the solution is begun.

Also, the boundary conditions for normal tube-side coolant channel flow are

$$f_1(1) = 1; \quad f_{2,3,4,5}(1) = 0; \quad f_6(1) = \Delta T_{\text{CT}}(1) \tag{5.16-43}$$

For reversed tube-side channel flow, they are

$$f_1(j\,\text{max}) = 1; \quad f_{2,3,4,5}(j\,\text{max}) = 0; \quad f_6(j\,\text{max}) = \Delta T_{\text{CT}}(j\,\text{max}) \tag{5.16-44}$$

and for both cases, they are

$$\begin{aligned} f_2(1) = 0; \quad f_4(1) = 0; \\ f_1(j\,\text{max}) = 0; \quad f_5(j\,\text{max}) = 0 \end{aligned} \tag{5.16-45}$$

5.16.2 Appendix 5.2: IHX Matrix Solution Algorithm

The solution of the matrix represented by Eq. (5.4-44) through Eq. (5.4-47) and whose coefficients are given in Section 5.16.1 is accomplished by Gaussian elimination, making use of the zeros present in the matrix. It is presented as an algorithm as it is coded in subroutine TSIHX. The arrows in the following mean replacement of what is on the left by the expression on the right.

JMAX = the number of nodes in the primary and intermediate coolant

JMAX $-$ 1 = the number of nodes in the shell and tube

1. Set j = 1.
2. Multiply Eq. (5.4-44) by $\frac{1}{a_1(j)}$

   $$a_2(j) \rightarrow \tfrac{a_2(j)}{a_1(j)}; a_3(j) \rightarrow \tfrac{a_3(j)}{a_1(j)};\ a_4(j) \rightarrow \tfrac{a_4(j)}{a_1(j)}; a_5 = \tfrac{a_5(j)}{a_1(j)};\ a_1(j) \rightarrow 1;$$

3. $e_1(j) \rightarrow e_1(j) - e_3(j)\, a_2(j)$

   $e_7(j) \rightarrow e_7(j) - e_3(j)\, a_3(j)$;

   $e_8(j) \rightarrow e_8(j) - e_8(j)\, e_4(j)$;

   $e_3(j) \rightarrow 0$;

4. Multiply non-zero coefficients in Eq. (5.4-45) by $1e_1(j)$.

   $$e_5(j) \longrightarrow \tfrac{e_5(j)}{e_1(j)};\ e_7(j) \longrightarrow \tfrac{e_7(j)}{e_1(j)};$$

   $$e_8(j) \longrightarrow \tfrac{e_8(j)}{e_1(j)};\ e_{10}(j) \longrightarrow \tfrac{e_{10}(j)}{e_1(j)}$$

   $$e_1(j) \longrightarrow 1$$

5. $c_1(j) \longrightarrow c_1(j) - c_2(j)\, e_5(j)$

   $c_3(j) \longrightarrow c_3(j) - c_2(j)\, e_7(j)$

   $c_4(j) \longrightarrow c_4(j) - c_2(j)\, e_{10}(j)$

   $c_6(j) \longrightarrow c_6(j) - c_2(j)\, e_8(j)$

   $c_2(j) \longrightarrow 0$

   $f_1(j) \longrightarrow f_1(j) - f_7(j)\, e_{10}(j)$

   $f_3(j) \longrightarrow f_3(j) - f_7(j)\, e_5(j)$

   $f_6(j) \longrightarrow f_6(j) - f_7(j)\, e_8(j)$

$$f_8(j) \longrightarrow f_8(j) - f_7(j)\, e_7(j)$$
$$f_7(j) \longrightarrow 0$$
$$e_1(j+1) \longrightarrow e_1(j+1) - e_6(j+1)\, e_7(j)$$
$$e_4(j+1) \longrightarrow e_4(j+1) - e_6(j+1)\, e_5(j)$$
$$e_8(j+1) \longrightarrow e_8(j+1) - e_6(j+1)\, e_8(j)$$
$$e_9(j+1) \longrightarrow e_9(j+1) - e_6(j+1)\, e_{10}(j)$$
$$e_6(j+1) \longrightarrow 0$$

6. Multiply non-zero coefficients in Eq. (5.4-46) by $\frac{1}{c_1(j)}$.

$$c_3(j) \longrightarrow \frac{c_3(j)}{c_1(j)}\ ;\ c_4(j) \longrightarrow \frac{c_4(j)}{c_1(j)}$$
$$c_5(j) \longrightarrow \frac{c_5(j)}{c_1(j)}\ ;\ c_6(j) \longrightarrow \frac{c_6(j)}{c_1(j)}$$
$$c_5(j) \longrightarrow \frac{c_5(j)}{c_1(j)}\ ;\ c_6(j) \longrightarrow \frac{c_6(j)}{c_1(j)}$$

7. $f_1(j) \longrightarrow f_1(j) - f_3(j)\, C_4(j)$
$f_8(j) \longrightarrow f_8(j) - f_3(j)\, C_3(j)$
$f_5(j) \longrightarrow f_5(j) - f_3(j)\, C_5(j)$
$f_6(j) \longrightarrow f_6(j) - f_3(j)\, C_6(j)$
$f_3(j) \longrightarrow 0$
$e_1(j+1) \longrightarrow e_1(j+1) - e_4(j+1)\, c_3(j)$
$e_9(j+1) \longrightarrow e_9(j+1) - e_4(j+1)\, c_4(j)$
$e_{10}(j+1) \longrightarrow e_{10}(j+1) - e_4(j+1)\, c_5(j)$
$e_8(j+1) \longrightarrow e_8(j+1) - e_4(j+1)\, c_6(j)$
$e_4(j+1) \longrightarrow 0$
$f_1(j+1) \longrightarrow f_1(j+1) - f_2(j+1)\, c_5(j)$
$f_4(j+1) \longrightarrow f_4(j+1) - f_2(j+1)\, c_4(j)$
$f_7(j+1) \longrightarrow f_7(j+1) - f_2(j+1)\, c_3(j)$
$f_6(j+1) \longrightarrow f_6(j+1) - f_2(j+1)\, c_6(j)$
$f_2(j+1) \longrightarrow 0$

8. Multiply non-zero coefficients in Eq. (5.4-47) by $\frac{1}{f_1(j)}$.

$$f_5(j) \longrightarrow \frac{f_5(j)}{f_1(j)}\ ;\ f_6(j) \longrightarrow \frac{f_6(j)}{f_1(j)};$$
$$f_8(j) \longrightarrow \frac{f_8(j)}{f_1(j)}\ ;\ f_1(j) \longrightarrow 1$$

9. $e_1(j+1) \longrightarrow e_1(j+1) - e_9(j+1)\, f_5(j)$
$e_{10}(j+1) \longrightarrow e_{10}(j+1) - e_9(j+1)\, f_5(j)$
$e_8(j+1) \longrightarrow e_8(j+1) - e_9(j+1)\, f_6(j)$
$e_9(j+1) \longrightarrow 0$
$f_1(j+1) \longrightarrow f_1(j+1) - f_4(j+1)\, f_5(j)$

$f_7(j+1) \longrightarrow f_7(j+1) - f_4(j+1) f_8(j)$

$f_6(j+1) \longrightarrow f_6(j+1) - f_4(j+1) f_6(j)$

$f_4(j+1) \longrightarrow 0$

10. Set $j \longrightarrow j+1$
11. If $j <$ JMAX, go to step 2
12. Multiply the non-zero coefficients in Eq. (5.4-47) by $\frac{1}{f_1(j)}$

    $f_6(j) \longrightarrow \frac{f_6(j)}{f_1(j)}$; $f_7(j) \longrightarrow \frac{f_7(j)}{f_1(j)}$; $f_1(j) \longrightarrow 1$
13. $e_1(j) \longrightarrow e_1(j) - e_{10}(j) f_7(j)$

    $e_8(j) \longrightarrow e_8(j) - e_{10}(j) f_6(j)$

    $e_{10}(j) \longrightarrow 0$
14. $e_8(j) \longrightarrow \frac{e_8(j)}{e_1(j)}$; $e_1(j) \longrightarrow 1$
15. $\Delta T_{\text{CP}}(j) = e_8(j)$

    $\Delta T_{\text{CT}}(j) = f_6(j) - f_7(j) \Delta T_{\text{CS}}(j)$
16. Set $j \longrightarrow j-1$
17. If $j < 1$, go to step 23
18. $\Delta T_{\text{CT}}(j) = f_6(j) - f_5(j) \Delta T_{\text{CT}}(j+1) - f_8(j) \Delta T_{\text{CS}}(j+1)$
19. $\Delta T_{\text{TU}}(j) = C_6(j) - C_3(j) \Delta T_{\text{CS}}(j+1) - C_4(j) \Delta T_{\text{CS}}(j+1) - C_5(j) \Delta T_{\text{CT}}(j+1)$
20. $\Delta T_{\text{CS}}(j) = e_8(j) - e_5(j) \Delta T_{\text{TU}}(j) - e_7(j) \Delta T_{\text{CS}}(j+1) - e_{10}(j) \Delta T_{\text{CT}}(j)$
21. $\Delta T_{\text{SH}}(j) = a_4(j) - a_2(j) \Delta T_{\text{CS}}(j) - a_3(j) \Delta T_{\text{CS}}(j+1)$
22. Go to step 16
23. End

5.16.3 Appendix 5.3: Cover Gas Flow and Pressure Algorithm

The main steps in the cover gas treatment in Section 5.7 may be summarized in the following algorithm:

1. Initialize. Set $I_{\text{n}} = 0$.
2. Start a loop on I_{o} for I_{o} = ICV1, …, ICV2. Set up INEW (I_{o}), IOLD (I_{n}), where I_{o} is the old compressible volume number and I_{n} is the new compressible volume number for the compressed arrays.
3. Is there gas in this compressible volume? Is ITYPCV (I_{o}) = 2,3,4,5, or 8? If so, go to step 5.
4. No gas in this compressible volume. INEW $(I_{\text{o}}) = 0$,

   $p_4(I_{\text{o}}) = p_3(I_{\text{o}})$,

   $T_4(I_{\text{o}}) = T_3(I_{\text{o}})$,

   $m_4(I_{\text{o}}) = 0$. Go to the end of the loop, step 6.
5. Gas in this compressible volume.

$I_{\text{n}} = I_{\text{n}} + 1$

INEW $(I_{\text{o}}) = I_{\text{n}}$

IOLD $(I_{\text{n}}) = I_{\text{o}}$

$p'_3 (I_{\text{n}}) = p_3 (I_{\text{o}}) , m'_3 (I_{\text{n}}) = m_3 (I_{\text{o}}) ,$

$V'_3 (I_{\text{n}}) = V_3 (I_{\text{o}}) , V'_4 (I_{\text{n}}) = V_4 (I_{\text{o}}) ,$

$\tau' (I_{\text{n}}) = \tau (I_{\text{o}})$

6. End of loop on I_{o}. Loop back to step 3 for the next value of I_{o}.
7. $I_{\text{nmax}} = I_{\text{n}}$
8. Start a loop on I_{n} for I_{n} = 1, …, I_{nmax}

Compute the adiabatic expansion and heat flow, as well as a temporary array $c\,(i)$ (see below).

9. Re-calculate $p'_3 (I_{\text{o}})$ giving.

   $p'_3 (I_{\text{n}}) = p'_3 (I_{\text{n}}) \left\{ 1 - \gamma \left[\frac{V'_4 (I_{\text{n}}) - V'_3 (I_{\text{n}})}{V'_4 (I_{\text{n}})} \right] \right\}$

   Calculate $T'_3 (I_{\text{n}})$.

   Set $c\,(I_{\text{n}}) = \frac{p'_3 (I_{\text{n}}) \gamma \Delta t_{\text{s}}}{m'_3 (I_{\text{n}}) T'_3 (I_{\text{n}})}$

   Also set $c\,(I, J) = 0$, with $c\,(I_{\text{n}}, I_{\text{n}}) = 0$,

   and $d\,(I_{\text{n}}) = 0$.
10. End of loop on I_{n}. Loop back to step 9 for the next value of I_{n}.
11. Is there more than one gas compressible volume? Is $I_{\text{nmax}} > 1$? If so, go to step 13.
12. Go to step 25.
13. Start loop on I for I = 1, …, $I_{\text{nmax}} - 1$.

Initialize e_{ij}, $F_{1\text{ij}}$, $F_{2\text{ij}}$.

14. Start loop on J for $J = I_{\text{n}} + 1$, …, I_{nmax}
15. Set $c\,(I, J) = 0.0$, $c\,(J, I) = 0.0$, $F_1\,(I, J) = 0.0$, $F_1\,(J, I) = 0.0$, $F_2\,(I, J) = 0.0$, $F_2\,(J, I) = 0.0$
16. End of loop on J. Loop back to step 11 for additional values of J.
17. End of loop on I. Loop back to step 14 for additional values of I.
18. Are there any gas segments? Is ISG1 > 0?

If not, go to step 25.

19. Start loop on I_{s} for I_{s} = ISG1, …, ISG2.

Compute gas flow between compressible volumes.

20. Find I_{ni} and I_{no}, the inlet and outlet compressible volumes for gas segment I_{s}.

    $I_{\text{ni}} =$ INEW (JNODG $(1, I_{\text{s}})$)

    $I_{\text{no}} =$ INEW (JNODG $(2, I_{\text{s}})$)
21. Iterate routine to obtain $F'_{\text{o}},\ F'_1,\ F'_2,\ T_{\text{ij}}$ (See remarks after Eq. (5.7-12)).

22. $F_1(I_{\text{ni}}, I_{\text{no}}) = F_1(I_{\text{ni}}, I_{\text{no}}) + F'_1$
    $F_1(I_{\text{no}}, I_{\text{ni}}) = F_1(I_{\text{no}}, I_{\text{ni}}) + F'_1$
    $F_2(I_{\text{ni}}, I_{\text{no}}) = F_2(I_{\text{ni}}, I_{\text{no}}) + F'_2$
    $F_2(I_{\text{no}}, I_{\text{ni}}) = F_2(I_{\text{no}}, I_{\text{nio}}) + F'_2$
    $c(I_{\text{no}}, I_{no}) = c(I_{\text{no}}, I_{\text{no}}) - e(I_{\text{no}}) T_{\text{ij}} F'_1$
    $c(I_{\text{no}}, I_{\text{ni}}) = c(I_{\text{no}}, I_{\text{ni}}) - e(I_{\text{no}}) T_{\text{ij}} F'_1$
    $c(I_{\text{ni}}, I_{\text{ni}}) = c(I_{\text{ni}}, I_{\text{ni}}) - e(I_{\text{ni}}) T_{\text{ij}} F'_2$
    $c(I_{\text{ni}}, I_{\text{no}}) = c(I_{\text{ni}}, I_{\text{no}}) - e(I_{\text{ni}}) T_{\text{ij}} F'_2$
    $d(I_{\text{no}}) = d(I_{\text{no}}) + e(I_{\text{no}}) T_{\text{ij}} F'_0$
    $d(I_{\text{ni}}) = d(I_{\text{ni}}) - e(I_{\text{ni}}) T_{\text{ij}} F'_0$
    $F_{\text{os}}(I_{\text{ni}}) = F_{\text{os}}(I_{\text{no}}) + F'_0$
    $F_{\text{os}}(I_{\text{ni}}) = F_{\text{os}}(I_{\text{no}}) - F'_0$
    $F_{\text{os}}(I_{\text{s}}) = F'_0$
    $F_{\text{s1}}(I_{\text{s}}) = F'_1$
    $F_{\text{s2}}(I_{\text{s}}) = F'_2$
23. End of loop on I_{s}. Loop back to step 20 for the next value of I_{s}.
24. Solve the matrix equation for $\Delta p(i)$ (see Eq. (5.7-20)).
25. Start loop on I_{n} for $I_{\text{n}} = 1, \ldots I_{\text{nmax}}$
26. $p'_4(I) = p'_3(I) + \Delta p_{\text{i}}$
    $m'_4(I) = m'_3(I) + \Delta t_{\text{x}} F_{\text{os}}(I)$
27. End of loop on I_{n}. Loop back to step 26 for additional values of I_{n}.
28. If $I_{\text{nmax}} = 1$, go to step 34.
29. Start loop on I for I = 1, …, $I_{\text{nmax}} - 1$.
30. Start loop on J for $J = I + 1$, I_{nmax}.
31. $m'_4(I) = m'_4(I) + \Delta t_{\text{s}} [F_1(J, I) \Delta p(I) + F_2(J, I) \Delta p(J)]$
    $m'_4(J) = m'_4(J) + \Delta t_{\text{s}} [F_1(I, J) \Delta p(J) + F_2(I, J) \Delta p(J)]$
32. End of loop on J. Loop back to step 31 for additional values of J.
33. End of loop on I. Loop back to step 30 for additional values of I.
34. For $I = 1$, …, I_{nmax} set
    $T'_4(I) = \frac{p'_4(I) V'_4(I)}{n'_4(I) R}$
35. Start loop on I_{n} for $I_{\text{n}} = 1$, …, I_{nmax}
36. $I_o = \text{IOLD}(I_n)$
    $p_3(I_o) = p'_3(I_n)$
    $m_3(I_o) = m'_3(I_n)$
    $T_3(I_o) = T'_3(I_n)$

37. End of loop on I_{n}. Loop back to step 36 for additional values of I_{o}.
38. If ISG1 $= 0$, go to step 40.
39. For $I_{\mathrm{s}} =$ ISG1, …, ISG2

    $I_1 =$ INEW (JNODG $(1, I_{\mathrm{s}})$)

    $I_2 =$ INEW (JNODG $(2, I_{\mathrm{s}})$)

    $F_{\mathrm{g4}}(I_{\mathrm{s}}) = F_{\mathrm{so}}(I_{\mathrm{s}}) + F_{\mathrm{s1}}(I_{\mathrm{s}})\,\Delta p(I_2) = F_{\mathrm{s2}}(I_{\mathrm{s}})\,\Delta p(I_1)$
40. Optional debugging print-out.
41. Return

5.16.4 Appendix 5.4: Air Blast Heat Exchanger Stack Momentum Equation

This appendix derives an expression for the air mass flowrate through the natural convection stack. The stack contains an opening at its base through which air is drawn in, the air passes over the finned tubes of the air blast heat exchanger and then rises to be exhausted at the top of the stack.

The one-dimensional steady-state momentum equation for flow in a channel of uniform cross section is

$$\frac{\mathrm{dP}}{\mathrm{dz}} = \frac{-\mathrm{d}}{\mathrm{dz}}\left(\rho v^2\right) - \rho g \sin\theta - \tau\frac{P_{\mathrm{w}}}{A}\,, \tag{5.16-46}$$

where

p =pressure

ρ =density

v =velocity

τ =wall shear stress

P_{w} =wetted perimeter

A =flow area

θ =channel inclination relative to horizontal

Integrating Eq. (5.16-46) gives the pressure change along the channel

$$\Delta P = -\left(\frac{w}{A}\right)^2\left(\frac{1}{\rho_{\mathrm{o}}} - \frac{1}{\rho_{\mathrm{i}}}\right) - \rho_{\mathrm{m}} g \updownarrow \sin\theta - \frac{K}{2\rho_{\mathrm{m}}}\left(\frac{w}{a}\right)^2 \tag{5.16-47}$$

where

w =channel mass flowrate

$\updownarrow$ =channel length

ρ_{o} =outlet density

ρ_{i} =inlet density

ρ_{m} =mean density

K =flow loss coefficient

Using Eq. (5.16-47), the pressure change form stack inlet to above the heat exchanger is

$$\Delta P = -\left(\frac{w}{A_{\mathrm{R}}}\right)^2 \left[\frac{K_{\mathrm{SI}}}{2\rho_{\mathrm{c}}}\left(\frac{A_{\mathrm{R}}}{A_{\mathrm{SI}}}\right)^2 + \frac{K_{\mathrm{HX}}}{2\rho_{\mathrm{c}}}\left(\frac{A_{\mathrm{R}}}{A_{\mathrm{HX}}}\right)^2\right] \tag{5.16-48}$$

where

A_{SI} = stack inlet cross-sectional area

A_{R} = riser cross-sectional area

A_{HX} = flow area at heat exchanger

K_{SI} = stack inlet loss coefficient

K_{HX} = heat exchanger loss coefficient

ρ_{c} =inlet air density

The gravity and acceleration terms have been neglected.

Similarly, the pressure change from the start of the riser to the stack outlet is

$$\Delta P = -\left(\frac{w}{A_{\mathrm{R}}}\right)^2 \frac{(K_{\mathrm{SO}} + K_{\mathrm{R}})}{2\rho_{\mathrm{h}}} - \rho_{\mathrm{h}} g \updownarrow \tag{5.16-49}$$

where

K_{SO} = stack outlet loss coefficient

K_{R} = riser loss coefficient

ρ_{h} = riser air density

$\updownarrow$ = riser length

The pressure change from the stack outlet through the outside air back to the stack inlet is approximately

$$\Delta P = \rho_{\mathrm{c}} g \updownarrow \tag{5.16-50}$$

The above three pressure changes, Eq. (5.16-48) through Eq. (5.16-50), must sum to zero since they are taken around a closed circuit. Solving for the air flowrate yields

$$w^2 = \frac{(\rho_{\mathrm{c}} - \rho_{\mathrm{h}})\, g \updownarrow A_{\mathrm{R}}^2}{\frac{K_{\mathrm{SI}}}{2\rho_{\mathrm{c}}}\left(\frac{A_{\mathrm{R}}}{A_{\mathrm{SI}}}\right)^2 + \frac{K_{\mathrm{HX}}}{2\rho_{\mathrm{c}}}\left(\frac{A_{\mathrm{R}}}{A_{\mathrm{HX}}}\right)^2 + \frac{(K_{\mathrm{SO}}+K_{\mathrm{R}})}{2\rho_{\mathrm{h}}}} \tag{5.16-51}$$

5.16.5 Appendix 5.5: Interpolation from User-Supplied Tables

A number of user-supplied tables are used in the SAS4A/SASSYS-1 code. These tables are of the form $Y\,(i)$ as a function of $X\,(i)$. The default option is to use linear interpolation between points in the table to obtain Y as a function of X. If X is between $X\,(J)$ and $X\,(J+1)$, then linear interpolation gives

$$Y\,(X) = Y\,(J) + \frac{Z\,[Y\,(J+1) - Y\,(J)]}{[X\,(J+1) - X\,(J)]} \tag{5.16-52}$$

where

$$Z = X - X\,(J) \tag{5.16-53}$$

Linear interpolation has the advantages that it is simple to do and it never exhibits the wild behavior and spurious oscillations that higher order interpolation schemes can give. On the other hand, linear interpolation has some disadvantages. One disadvantage is that if only a few widely-spaced points are used for a table, linear interpolation may not be very accurate between points. Another disadvantage is that linear interpolation gives slope discontinuities at the table points. These slope discontinuities can cause corresponding slope discontinuities in plots of powers, temperatures, and flow rates calculated by the code. Sometimes these slope discontinuities correspond to real physical events, such as scramming the control rods or tripping the pumps; but usually they are numerical artifacts of the interpolation scheme. In order to address this problem, alternative interpolation schemes are available for some of the tables used by the code. Table 5.16.1 lists the interpolation options.

For the third order fit,

$$Y(X) = C_{1J} + C_{2J}Z + C_{3J}Z^2 + C_{4J}Z^3 \tag{5.16-54}$$

where the coefficients C are picked such that:

a) Y is continuous at $X(J)$ and $X(J+1)$.

b) The slope of Y is continuous at internal points.

c) The value of F is minimized, where

$$F = \int_{X_{min}}^{X_{max}} \left(\frac{d^2y}{dx^2}\right)^2 dx \tag{5.16-55}$$

Table 5.16.1: Table Interpolation Options for a Table of $Y(i)$ vs. $X(i)$, as Determined by the User-Specified Parameter IFIT

IFIT $(K) = 0$	Use linear fits
1	3rd order fits
2	3rd order fits with slope discontinuities at points where $X(i+1) - X(i) < .001$
3	Linear fit to $\log(Y)$
4	3rd order fit to $\log(Y)$
5	3rd order fit to $\log(Y)$ with slope discontinuities
$K = 1$	Power vs time or user specified reactivity vs time
2	PRIMAR-1 pump head vs time or channel flow vs time
3	PRIMAR-1 inlet temperature vs time
4	PRIMAR-4 pump head, motor torque, or pump speed vs time

Note: For all other tables, linear interpolation is used.

In this equation X_{min} and X_{max} are the first and last points in the table if IFIT = 1 or 4. If IFIT = 2 or 5, then the table is broken up into ranges, with the boundaries between ranges occurring at the points where the slope discontinuities occur. In this case, the fitting is done separately for each range; and X_{min} and X_{max} are the points at the ends of the range. There are some special cases or exceptions:

1. For all fitting options, if X is less than the first entry in the table, then Y is set equal to the first value of Y in the table. Also, if X is greater than the last entry in the table, then the last value of Y is used.
2. For IFIT = 2 or 5, in an interval where a slope discontinuity occurs $(X(i+1) - X(i) < .001)$ linear interpolation is used from $X(i)$ to $X(i+1)$.
3. If there are only two points in a range, then linear interpolation is used.
4. For IFIT = 1, 2, 4 or 5, if there are only 3 points in a range, then a parabola is fitted to the 3 points to determine the values of C in the range. Third order fits are only used if there are at least 4 points in a range.

Care should be taken when using third order fits, since they can result in spurious wiggles. Figure 5.16.1 shows the results of a third order fit to the power level in the SHRT-17 test in EBR-II. In this case, the third order fit to the power produces wiggles, whereas the third order fit to the logarithm of the power produces a smooth curve.

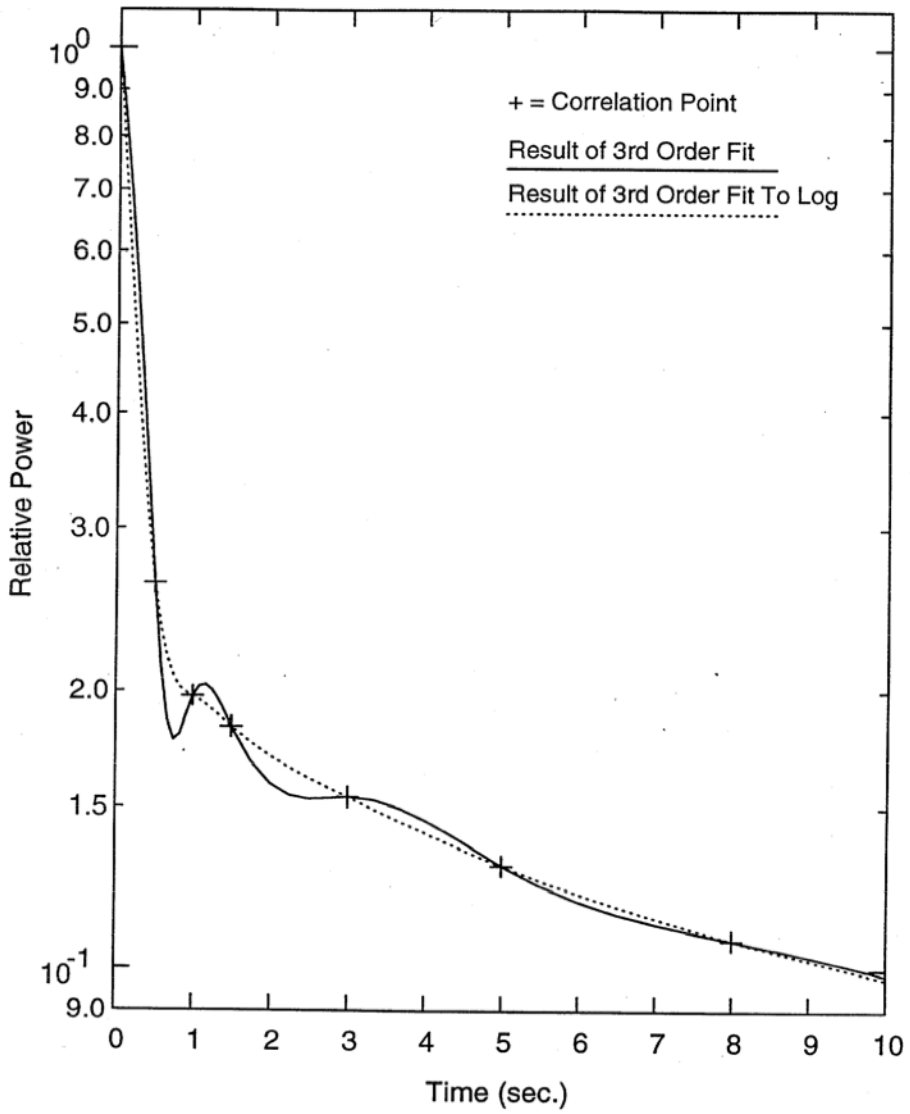


Figure 5.16.1: Figure A5.5-1. Fit to SHRT-17 Normalized Total Power, Early Times.

5.16.6 Appendix 5.6: Optional Eulerian Solution for Pipe Temperatures

As mentioned in Section 5.4.1, a Eulerian calculation can be used to speed up the pipe temperature calculation if the coolant moves more than two nodes in a time step. This Eulerian speed-up has only been implemented for flow in the nominal direction; if flow reversal has occurred in a pipe, then the Eulerian calculation is not used.

For the Eulerian calculation, Eq. (5.4-1) for the coolant is replaced by

$$\rho_{\mathrm{c}} c_{\mathrm{c}} A_{\mathrm{c}} \frac{\partial \mathrm{T_c}}{\partial \mathrm{t}} + w c_{\mathrm{c}} \frac{\partial \mathrm{T_c}}{\partial \mathrm{z}} = P_{\mathrm{er}} h_{\mathrm{wc}} \left(T_{\mathrm{w}} - T_{\mathrm{c}}\right) + q'_c \tag{5.16-56}$$

Eq. (5.4-2) is still used for the wall. Finite differencing of Eq. (5.16-56) gives

$$\begin{aligned} &\rho_{\mathrm{c}} c_{\mathrm{c}} A_{\mathrm{c}} \left[\frac{T_{\mathrm{c6j}} + T_{\mathrm{c6j+1}} - T_{\mathrm{c5j}} - T_{\mathrm{c5j+1}}}{2\delta t}\right] \\ &+\frac{w c_{\mathrm{c}}}{\Delta z}\left[\theta_1 \left(T_{\mathrm{c5j+1}} - T_{\mathrm{c5j}}\right) + \theta_2 \left(T_{\mathrm{c6j+1}} - T_{\mathrm{c6j}}\right)\right] = \\ &P_{\mathrm{er}} h_{\mathrm{wc}} \left\{\theta_1 \left[T_{\mathrm{w5j}} - \frac{T_{\mathrm{c5j}} + T_{\mathrm{c5j+1}}}{2}\right]\right. \\ &\left.+\theta_2 \left[T_{\mathrm{w6j}} - \frac{T_{\mathrm{c6j}} + T_{\mathrm{c6j+1}}}{2}\right]\right\} + q'_{cj} \end{aligned} \tag{5.16-57}$$

Similarly, finite differencing of Eq. (5.4-2) gives

$$\begin{aligned} \frac{M_{\mathrm{w}} C_{\mathrm{w}}}{\delta t}\left(T_{\mathrm{w6j}} - T_{\mathrm{w5j}}\right) = P_{\mathrm{er}} h_{\mathrm{wc}} &\left[\theta_1 \left(\frac{T_{\mathrm{c5j}} + T_{\mathrm{c5j+1}}}{2} - T_{\mathrm{w5j}}\right)\right. \\ &\left.+\theta_2 \left(\frac{T_{\mathrm{c6j}} + T_{\mathrm{c6j+1}}}{2} - T_{\mathrm{w6j}}\right)\right] \\ &+ \left(hA\right)_{\mathrm{snk}} \left(T_{\mathrm{snk}} - \theta_1 T_{\mathrm{w5j}} - \theta_2 T_{\mathrm{w6j}}\right) + q'_{wj} \end{aligned} \tag{5.16-58}$$

Eq. (5.16-58) can be rewritten as

$$T_{\mathrm{w6j}} = B_{\mathrm{w0j}} + B_{\mathrm{w1j}} \left(T_{\mathrm{c6j}} + T_{\mathrm{c6j+1}}\right) \tag{5.16-59}$$

where

$$B_{\mathrm{w1j}} = \frac{P_{\mathrm{er}} h_{\mathrm{wc}} \theta_2 \delta t}{2 d_{\mathrm{w}}} \tag{5.16-60}$$

$$d_{\mathrm{w}} = M_{\mathrm{w}} C_{\mathrm{w}} + \theta_2 \delta t \left[P_{\mathrm{er}} h_{\mathrm{wc}} + \left(hA\right)_{\mathrm{snk}}\right] \tag{5.16-61}$$

and

$$\begin{aligned} B_{\mathrm{w0j}} = \Bigg\{ &M_{\mathrm{w}} C_{\mathrm{w}} T_{\mathrm{w5j}} + P_{\mathrm{er}} h_{\mathrm{wc}} \theta_1 \delta t \left[\frac{T_{\mathrm{c5j}} + T_{\mathrm{c5j+1}}}{2} - T_{\mathrm{w5j}}\right] \\ &+ \left(hA\right)_{\mathrm{snk}} \delta t \left(T_{\mathrm{snk}} - \theta_1 T_{\mathrm{w5j}}\right) + q'_{wj} \delta t \Bigg\} / d_{\mathrm{w}} \end{aligned} \tag{5.16-62}$$

Similarly, Eq. (5.16-57) can be rewritten as

$$T_{\mathrm{c6j+1}} = B_{\mathrm{c0j}} + B_{\mathrm{c1j}} T_{\mathrm{c6j}} + B_{\mathrm{c2j}} T_{\mathrm{w6j}} \tag{5.16-63}$$

where

$$\begin{aligned} B_{\mathrm{c0j}} = \Big\{ &\Delta z \rho_{\mathrm{c}} c_{\mathrm{c}} A_{\mathrm{c}} \left(T_{\mathrm{c5j}} + T_{\mathrm{c5j+1}}\right) + 2\theta_1 \delta t w c_{\mathrm{c}} \left(T_{\mathrm{c5j}} - T_{\mathrm{c5j+1}}\right) \\ &+2\theta_1 \delta t \Delta z P_{\mathrm{er}} h_{\mathrm{wc}} \left[T_{\mathrm{w5j}} - \frac{T_{\mathrm{c5j}} + T_{\mathrm{c5j+1}}}{2}\right] + 2\delta t \Delta z q'_{cj} \Big\} / d_{\mathrm{c}} \end{aligned} \tag{5.16-64}$$

$$d_{\mathrm{c}} = \rho_{\mathrm{c}} c_{\mathrm{c}} A_{\mathrm{c}} \Delta z + 2\theta_2 \delta t \left(w c_{\mathrm{c}} + \frac{P_{\mathrm{er}} h_{\mathrm{wc}}}{2} \Delta z \right) \tag{5.16-65}$$

$$B_{\mathrm{c1j}} = - \left\{ \rho_{\mathrm{c}} c_{\mathrm{c}} A_{\mathrm{c}} \Delta z + 2\theta_2 \delta t \left(\frac{P_{\mathrm{er}} h_{\mathrm{wc}}}{2} \Delta z - w c_{\mathrm{c}} \right) \right\} / d_{\mathrm{c}} \tag{5.16-66}$$

and

$$B_{\mathrm{c2j}} = \frac{2\theta_2 \delta t \Delta z P_{\mathrm{er}} h_{\mathrm{wc}}}{d_{\mathrm{c}}} \tag{5.16-67}$$

Eq. (5.16-59) and Eq. (5.16-63) can be combined to give

$$T_{\mathrm{c6j+1}} = B_{\mathrm{cc0j}} + B_{\mathrm{cc1j}} T_{\mathrm{c6j}} \tag{5.16-68}$$

where

$$B_{\mathrm{cc0j}} = \frac{B_{\mathrm{c0j}} + B_{\mathrm{c2j}} B_{\mathrm{w0j}}}{1 - B_{\mathrm{c2j}} B_{\mathrm{w1j}}} \tag{5.16-69}$$

and

$$B_{\mathrm{cc1j}} = \frac{B_{\mathrm{c1j}} + B_{\mathrm{c2j}} B_{\mathrm{w1j}}}{1 - B_{\mathrm{c2j}} B_{\mathrm{w1j}}} \tag{5.16-70}$$

Note that B_{cc0j} and B_{cc1j} can be calculated before the temperatures at the end of the sub-interval are known.

The pipe inlet temperature at the end of the step is used to set the first coolant temperature, T_{c61}. The code marches along the pipe, using Eq. (5.16-68) to calculate $T_{\mathrm{c6j+1}}$ after T_{c6j} has been calculated. After the coolant temperatures have been calculated, Eq. (5.16-59) is used to calculate the wall temperatures.

5.16.7 Appendix 5.7: Detailed Derivation of EQ EM Pump

The equivalent circuits for the detailed and the simple EM pump models can be reduced to the circuit shown in Figure 5.16.2, where Z_i is the impedance of each component in the path of the total phase current, I, being driven by the phase voltage, V_s.

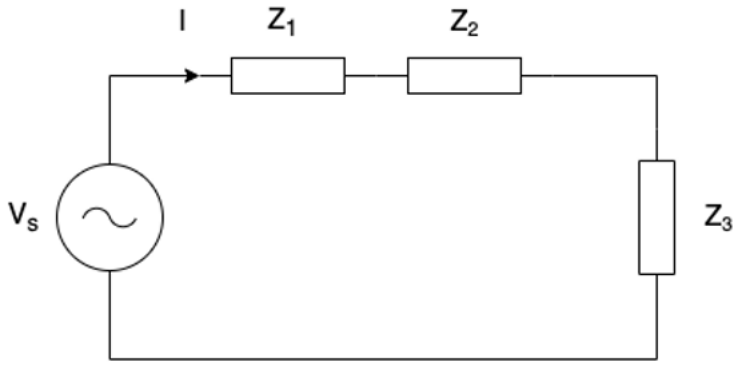


Figure 5.16.2: Reduced Equivalent Circuit for the EM pump models.

Because the equivalent circuits contain inductors, the circuit's total impedance is a complex number.

$$\begin{aligned} Z_T &= Z_1 + Z_2 + Z_3 \\ &= Re\,(Z_T) + Im(Z_T) j \\ &= Re\,(Z_1) + Re\,(Z_2) + Re\,(Z_3) + Im\,(Z_1)\, j + Im\,(Z_2)\, j + Im\,(Z_3)\, j \end{aligned} \tag{5.16-71}$$

Using the total impedance, the total phase current can be quantified using Ohm's law.

$$\begin{aligned} I &= \frac{V}{Z_T} \\ |I| &= \frac{|V|}{|Z_T|} \\ |I| &= \frac{|V|}{\sqrt{\mathrm{Re}\left(Z_T\right)^2 + Im\left(Z_T\right)^2}} \end{aligned} \tag{5.16-72}$$

The presence of the inductors introduces a phase difference between the phase voltage and current. The phase difference is defined as

$$\tan\left(\phi\right) = \frac{\mathrm{Im}\left(Z_T\right)}{Re(Z_T)} \tag{5.16-73}$$

5.16.7.1 Simple Circuit Analysis

For the simple EM pump circuit, the impedance of each component is

$$Z_1 = R_1 \tag{5.16-74}$$

$$Z_2 = X_1 j \tag{5.16-75}$$

$$Z_3 = \left(\frac{1}{X_2 j} + \frac{s}{R_2}\right)^{-1} = \frac{R_2 X_2 j}{R_2 + \mathrm{sX}_2 j} = \frac{s R_2 X_2^2}{R_2^2 + s^2 X_2^2} + \frac{R_2^2 X_2}{R_2^2 + s^2 X_2^2} j \tag{5.16-76}$$

Using Eq. (5.16-71) and Eq. (5.16-72), the current as a function of slip, frequency and voltage can be determined.

$$|I| = \frac{|V|}{\sqrt{\left(R_1 + \frac{s R_2 X_2^2}{R_2^2 + s^2 X_2^2}\right)^2 + \left(X_1 + \frac{R_2^2 X_2}{R_2^2 + s^2 X_2^2}\right)^2}} \tag{5.16-77}$$

The current's dependence on slip is inherent, as $X_i = 2\pi L_i f$.

Ohm's law can be used to determine the division of the total current between the X_2 and R_2/s branches

$$|I|\,|Z_3| = |I_1|\,|X_2| = |I_2|\,\frac{R_2}{s} \tag{5.16-78}$$

Therefore, the power dissipated in R_2/s can be defined as

$$\begin{aligned} P_{R2} &= 3\,|I_2|\,|V_{R2}| \\ &= 3\frac{|I_2|^2\,R_2}{s} \end{aligned} \tag{5.16-79}$$

If we assume that a fraction of the power is lost to the fluid, the mechanical power dissipated in R_2/s is

$$\begin{aligned} P_{R2} &= 3\frac{|I_2|^2 R_2 (1-s)}{s} \\ &= 3|I|^2 |Z_3|^2 (1-s)\frac{R_2}{s} \\ &= 3|I|^2 \left(\frac{|Z_3|}{\frac{R_2}{s}}\right)^2 (1-s)\frac{R_2}{s} \\ &= 3\frac{s^4 X_2^4 + s^2 R_2^2 X_2^2}{\left(R_2^2 + s^2 X_2^2\right)^2}(1-s)\frac{R_2}{s}|I|^2 \\ &= 3\frac{s^2 X_2^2}{\left(R_2^2 + s^2 X_2^2\right)}(1-s)\frac{R_2}{s}|I|^2 \\ &= 3|I|^2 \frac{R_2(1-s)}{\mathrm{s}\left(\frac{R_2^2}{s^2 X_2^2}+1\right)} \end{aligned} \tag{5.16-80}$$

Using the relationship between pumping power and pump head, the developed pump head for the simple EM pump model can be described as

$$\begin{aligned} \Delta P_{\text{pump}} &= \frac{P_{R2}}{Q} \\ &= 3|I|^2 \frac{R_2(1-s)}{\mathrm{Qs}\left(\frac{R_2^2}{s^2 X_2^2}+1\right)} \end{aligned} \tag{5.16-81}$$

5.16.7.2 Detailed Circuit Analysis

For the detailed EM pump circuit, the impedance of each component is

$$Z_1 = R_1 \tag{5.16-82}$$

$$Z_2 = X_1 j \tag{5.16-83}$$

$$\begin{aligned} Z_3 &= \left(\frac{1}{X_2 j} + \frac{1}{R_D} + \frac{1}{R_J + R_m}\right)^{-1} \\ &= \frac{(R_J + R_M) R_D X_2 j}{(R_J + R_m) R_D + (R_J + R_M + R_D) X_2 j} \\ &= \frac{(R_J + R_m)(R_J + R_m + R_D) R_D X_2^2 + (R_J + R_m)^2 R_D^2 X_2 j}{(R_J + R_m)^2 R_D^2 + (R_J + R_m + R_D)^2 X_2^2} \end{aligned} \tag{5.16-84}$$

The magnitude of Z_3 is

$$\begin{aligned} |Z_3|^2 &= \frac{(R_J + R_m)^2 (R_J + R_m + R_D)^2 R_D^2 X_2^4 + (R_J + R_m)^4 R_D^4 X_2^2}{\left((R_J + R_m)^2 R_D^2 + (R_J + R_m + R_D)^2 X_2^2\right)^2} \\ &= \frac{(R_J + R_m)^2 R_D^2 X_2^2}{(R_J + R_m)^2 R_D^2 + (R_J + R_m + R_D)^2 X_2^2} \end{aligned} \tag{5.16-85}$$

Using a similar analysis as was presented for the simple model, the current passing through the fluid is determined to be

$$|I|\,|Z_3| = |I_3|\,(R_J + R_m) \tag{5.16-86}$$

and the mechanical power delivered to the fluid is

$$\begin{aligned} P_m &= 3\left|I_3\right|^2 R_m \\ &= 3\frac{\left|I\right|^2\left|Z_3\right|^2}{\left(R_J+R_m\right)^2}R_m \\ &= 3I^2R_m\frac{R_D^2X_2^2}{\left(R_J+R_m\right)^2R_D^2+\left(R_J+R_m+R_D\right)^2X_2^2} \end{aligned} \tag{5.16-87}$$

Therefore, the developed pump head in the detailed EM pump model is

$$\begin{aligned} \Delta P &= \frac{P_m}{Q} \\ &= 3\frac{I^2R_m}{Q}\frac{R_D^2X_2^2}{\left(R_J+R_m\right)^2R_D^2+\left(R_J+R_m+R_D\right)^2X_2^2} \end{aligned} \tag{5.16-88}$$

5.16.7.3 Simple Circuit Parameters

In order to determine the resistance and inductance of the different components in the simple EM pump model, measurable quantities are provided at two distinct pump states. The first pump state, referred to as the rated state, provides a pump head, ΔP_R, measured flow rate, Q_R, measured phase voltage, V_R, measured phase current, I_R, measured slip, s_R, measured efficiency, η_R, and measured frequency, f_R. The second pump state, referred to as the stalled state, provides the pump head at stalled conditions, ΔP_S, for the measured frequency and measured phase voltage. $\Delta P_{P,R}$ and $\Delta P_{S,R}$ are functions of the measured pump head, ΔP_R, and measured pump head at stalled conditions, ΔP_S.

When IDPOPT = 0, then the user-provided ΔP_R and ΔP_S *include* pressure changes due to friction losses across the pump. When IDPOPT = 1, then the user-provided ΔP_R and ΔP_S *do not include* pressure changes due to friction losses across the pump.

To solve the initial conditions for the simplified pump model, the following terms are defined.

$$\Delta P_{P,S} = \Delta P_S \tag{5.16-89}$$

$$\Delta P_{P,R} = \begin{cases} \Delta P_R, & \text{IDPOPT} \neq 0 \\ \Delta P_R + \Delta P_{f,R}, & \text{IDPOPT} = 0 \end{cases}$$

where $\Delta P_{f,R}$ is the frictional pressure drop that occurs within the pump at the rated flow rate and IDPOPT is a user-defined flag. $\Delta P_{f,R}$ is internally calculated by SAS4A/SASSYS-1 using the geometry input.

In order to determine the 4 circuit parameters and the current at stalled conditions, five equations are required. Ohm's law provides us with two of the five equations:

$$\left(\frac{V_R}{I_R}\right)^2 = \left|Z_T(s=s_R)\right|^2 \tag{5.16-90}$$

$$\left(\frac{V_R}{I_S}\right)^2 = \left|Z_T(s=1)\right|^2 \tag{5.16-91}$$

The rated and stall pressures provide an additional two equations:

$$\Delta P_{P,R} = \frac{3I_R^2R_2}{Av_s\left(\frac{R_2^2}{X_2^2s_R}+s_R\right)} \tag{5.16-92}$$

$$\Delta P_{P,S} = \frac{3I_s^2 R_2}{Av_s\left(\frac{R_2^2}{X_2^2}+1\right)} \tag{5.16-93}$$

The final equation comes from the total power dissipated by the pump

$$P_R = 3V_R I_R \cos(\phi) \tag{5.16-94}$$

$$\cos(\phi) = \frac{\text{Re}\left(Z_T\left(s=s_R\right)\right)}{\left|Z_T\left(s=s_R\right)\right|} \tag{5.16-95}$$

$$P_R = \frac{\left(\Delta P_R\right) Q_R}{\eta_R} \tag{5.16-96}$$

$$Re\left(Z_T(s=s_R)\right)^2 = \left(\frac{\left(\Delta P_R\right) Q_R}{3I_R^2 \eta_R}\right)^2 \tag{5.16-97}$$

Note, the synchronous speed of the EM pump can be determined using the definition of slip and the rated conditions

$$v_s A = \frac{Q_R}{1-s_R} \tag{5.16-98}$$

Instead of solving the five equations at once, an iterative approach is taken. By assuming a stall current, Eq. (5.16-93) can be solved

$$\frac{R_2^2}{X_2^2} = \frac{3I_s^2 R_2}{Av_s \Delta P_s} - 1 \tag{5.16-99}$$

and plugged into Eq. (5.16-92), providing an equation for R_2

$$R_2 = \frac{\left(\Delta P_{P,R}\right)\left(Av_s \frac{\left(s_R^2-1\right)}{s_R}\right)}{3I_R^2 - \frac{3I_S^2}{\Delta P_{P,S} s_R}\Delta P_{P,R}} \tag{5.16-100}$$

with the value of R_2 known, X_2 can be determined using Eq. (5.16-99) and Eq. (5.16-97) can be solved to determine the value of R_1

$$R_1 = \frac{\left(\frac{\Delta P_R Q_R}{3I_R^2 \eta_R}\left(R_2^2 + X_2^2 {s_R}^2\right) - s_R R_2 X_2^2\right)}{\left(R_2^2 + X_2^2 s_R^2\right)} \tag{5.16-101}$$

With R_1, R_2, and X_2 known Eq. (5.16-90) can be used to determine X_1

$$X_1 = \left[\left(\frac{V_R}{I_R}\right)^2 - \text{Re}\left(Z_T(s=s_R)\right)^2\right]^{1/2} - \frac{R_2^2 X_2}{\left(R_2^2 + X_2^2 s_R^2\right)} \tag{5.16-102}$$

Finally, Eq. (5.16-91) will be solved to determine the error in the stall current. Using error in the stall current, a new stall current will be estimated and Eq. (5.16-99) through Eq. (5.16-102) will be repeated. This process continues until the error in the stall current is within a user defined limit.

CHAPTER

SIX

CONTROL SYSTEM

6.1 Introduction

The SAS4A/SASSYS-1 control system model was developed for the design and analysis of control systems in LMR plants. The model is described in Ref. 6-1 while application of the model is described in Ref. 6-2. In this chapter, the model is described and guidelines for using the model are given. Essentially, the user should be able to set up an input deck and run the model using the material in this section and in Section 6.8.

The model is very flexible, allowing the user to select any number of plant variables for input to the control system as measured quantities. These signals can then be processed by a user defined network of mathematical blocks that implement the control equations. The output from these blocks can then be used to drive various actuators already existing in SAS4A/SASSYS-1 or they can be used to directly control plant variables in SAS4A/SASSYS-1. The model has a steady- state solution finder that can be used to determine initial values for demand signals and state variables that place the control system in a steady state that is consistent with the plant steady state as calculated by SAS4A/SASSYS-1. The control system model can also be used to calculate auxiliary variables and print their values.

The model is an integral component of SAS4A/SASSYS-1 and is accessed through the input deck in a manner similar to the other reactor component models. Before using the model however, one must write the mathematical equations that describe the desired plant control system and identify the plant variables that are to be measured and controlled. The user then transforms the equations and variables into a block diagram where the individual component blocks are basic mathematical elements such as integrators and summers. The input deck is prepared directly from this block diagram with each block definition occupying an input card and each plant variable that links with the control system also occupying an input card. Several other cards must also be entered to specify how the control system initial conditions are to be calculated and to assign values to parameters that control the accuracy and stability of the transient solution. A set of parameters also exists for controlling the printing of debug data. This output is useful for diagnosing input errors.

This section describes the basic model; it also gives some general guidelines for using the model. The section assumes the reader has a knowledge of power plant control systems and is able to write the equations that describe their system. The organization of the material is as follows. In Section 6.2 the general equation form that can be represented is given. It is very probable that the user's model fits this form but this should be verified. The solution techniques used to solve the block diagram equations are described in Section 6.3. Section 6.4 presents some general guidelines for selecting values of solution control parameters and describes some of the model features and how they are used. The input description is given in Section 6.8.

6.2 Generalized Model

The control system model was developed with the intent that a wide range of plant control systems can be simulated. For this purpose, two specific objectives were set. First, the model should be general enough to permit the user to assemble any set of control equations and specify how they interface to the plant solely through the input. And second, the model should employ a numerical method which is reliable in all foreseeable applications. Fulfilling these two goals led to the identification of a general equation form capable of representing all classes of plant control systems.

6.2.1 General Equations

The solution algorithms of the model are based on a general set of equations for the control system state variables and outputs. These equations are formulated under the assumption that the three components of a control system, the sensor, the controller, and the actuator, can all be modeled as ordinary differential equations. The general equation form is easy to deduce.

Since the sensor and actuator behavior are governed by physical laws and they are normally modeled in lumped parameter form, they are both described by

$$\begin{aligned} \frac{\mathrm{d}}{\mathrm{dt}} x\left(t\right) &= f\left(x\left(t\right), u\left(t\right)\right) \\ y\left(t\right) &= g\left(x\left(t\right), u\left(t\right)\right) \end{aligned} \tag{6.2-1}$$

where

$x\left(t\right)$ = n x 1 state vector;

$u\left(t\right)$ = r x 1 input vector; and

$y\left(t\right)$ = m x 1 output vector.

The controller also has the basic form of Eq. (6.2-1) as it consists of integrating and function elements. But in addition a derivative element is sometimes used in which case derivatives appear on the right hand side of Eq. (6.2-1). In practice the output signal from an integrator will be differentiated at most once so that the controller equation is

$$\begin{aligned} \frac{\mathrm{d}}{\mathrm{dt}} x\left(t\right) &= f\left(x\left(t\right), \frac{\mathrm{d}}{\mathrm{dt}} x, u\left(t\right)\right) \\ y\left(t\right) &= g\left(x\left(t\right), u\left(t\right)\right) \ . \end{aligned} \tag{6.2-2}$$

The general equation form results when the equations for the three components are coupled and the signals that link to the plant are explicitly labeled

$$\begin{aligned} \frac{\mathrm{d}}{\mathrm{dt}} x\left(t\right) &= f\left(x\left(t\right), \frac{\mathrm{d}}{\mathrm{dt}} x, u_{\text{mea}}\left(t\right), u_{\text{dmd}}\left(t\right)\right) \\ y_{\text{ctl}}\left(t\right) &= g\left(x\left(t\right), u_{\text{mea}}\left(t\right), u_{\text{dmd}}\left(t\right)\right) \ , \end{aligned} \tag{6.2-3}$$

where

$u_{\text{mea}}\left(t\right) = 1 \times n_{\text{mea}}$ measured input vector;

$u_{\text{dmd}}\left(t\right) = 1 \times n_{\text{dmd}}$ demand input vector; and

$y_{\text{ctl}}\left(t\right) = 1 \times n_{\text{ctl}}$ control system output vector.

To guide the choice of initial conditions and their calculation for the above equations, we must consider the intended applications. Since the code is ultimately to be used for analysis of plant wide transients, the initial

conditions must be compatible with the way in which these transients begin. Generally the user prescribes the plant steady state and therefore it should be reasonable to initialize the control system so that at time zero it preserves this steady state. In this case boundary conditions for the control system are taken from the plant, and control system time derivatives are set to zero. Writing the control equations explicitly in terms of the measured signals, control signals and the demand signals, Eq. (6.2-3) becomes

$$\begin{aligned} 0 &= f\left(x\left(0\right), 0, u^{*}{}_{\text{mea}}\left(0\right), u_{\text{dmd}}\left(0\right)\right) \\ 0 &= g\left(x\left(0\right), u^{*}{}_{\text{mea}}\left(0\right), u_{\text{dmd}}\left(0\right)\right) - y^{*}{}_{\text{ctl}}\left(0\right) \ , \end{aligned} \tag{6.2-4}$$

where

$y^{*}{}_{\text{ctl}}\left(0\right) = 1 \times n_{\text{ctl}}$ vector of plant values associated with $y_{\text{ctl}}\left(0\right)$; and

$u^{*}{}_{\text{mea}}\left(0\right) = 1 \times n_{\text{mea}}$ vector of plant values associated with $u_{\text{mea}}\left(0\right)$.

The asterisk denotes steady state conditions in the plant. The initial conditions then that place the control system in steady state equilibrium with the plant are the values of $u_{\text{dmd}}\left(0\right)$ and $x\left(0\right)$ that satisfy Eq. (6.2-4).

6.2.2 Block Diagram Approach

One might well ask what benefits can be obtained from a knowledge of this general equation. The principal benefit is a flexible modeling approach that permits the user to describe the plant control equations in a block diagram manner. The key to achieving this capability is the fact that the properties of the general equation form are well known and can be brought to bear on the development of a reliable numerical scheme.

The process by which the user describes their block diagram is analogous to the process of programming an analog computer. Basically, four types of information must be supplied. First, each mathematical block must be defined and the interconnections among them specified. Presently there are twenty-three blocks to choose from and these are shown in Table 6.8.2; additional blocks can be added if required. Each block can accept up to two signals at its input for processing and supply the result, termed a block signal, for further processing by other blocks. Second, each forcing function driving the collection of blocks must be defined. A demand signal is available for this purpose and references a user-supplied table of values that specifies the signal as a function of time. Third, plant measured quantities input to the collection of blocks must be defined. A measured signal is available for this purpose and permits access to a number of plant variables including temperature, flow, pressure and inventory in a number of reactor components. A complete list is given in Table 6.8.3. Finally, those block signals that are used to drive the plant must be defined. For that purpose, control signals can be defined by the user to represent, among other things, sources of external reactivity, feedwater mass flowrate and pump motor torque. A complete list is given in Table 6.8.3.

6.3 Solution Techniques

So far we have focused on the benefits derived from a generalized model, but have not touched on the methods used to implement the model. We will now describe the numerical techniques, first discussing the potential problems that can occur when solving a set of equations of the form Eq. (6.2-3) and then describing the numerical methods used to handle them.

Because the model is generalized, the solution techniques should be transparent for a wide range of situations that can arise. In the case of the steady-state solution finder, it is clear that the equations and variables to be solved for are given by Eq. (6.2-4). In certain instances these equations may not be square yet a solution exists while at other times the Jacobian of the right hand side may be singular. In the case of the transient solution it is important that the solution technique be able to maintain a user specified level of solution accuracy under a wide range of system response times. Solution techniques capable of handling these situations might be termed robust. We describe such techniques here.

The reader is cautioned that the difference equations we write may appear unconventional. The equations of reactor physics and thermal hydraulics when differenced in one-dimensional space appear with a single index denoting location in space. However in the generalized control system problem, geometry or space does not enter. Instead the continuum of space is replaced by a logical relationship among block outputs, block inputs and plant signals. These relationships must be recorded as part of the specific problem definition and, as will become clear in this work, vectors can be used to store these relationships. To summarize, the usual physical relationships that exist between points in space in conventional difference equations are replaced by logical relationships among blocks.

6.3.1 Definition of Block Diagram

The specification of the user's control equations and the solution of these are closely tied to the block diagram concept. Therefore, before the solution techniques can be discussed, we need to define the block diagram.

The block diagram is represented through several vectors whose entries define the types of blocks and the interconnections among blocks. The vectors are one-dimensional and the index to the elements can be thought of as analogous to a space index. To begin with, a unique signal number is assigned to the output of a block. If there are n_{blk} blocks occurring in the user's input and the i-th one is assigned signal number m and the block is of type k then the following entries are created

$$L_{\text{blk,i}} = m \quad \text{and} \quad K_{\text{blk,m}} = k; \quad i = 1, ..., n_{\text{blk}} \ .$$

Further if the inputs to this block are signal q and r then the entries

$$J1_{\text{blk,m}} = q \quad \text{and} \quad J2_{\text{blk,m}} = r$$

are also created. If the block is a non-dynamic one then the block operator is

$F_{\text{k}} \left(S_{\text{q}}^{j}, S_{\text{r}}^{j} \right)$ = value of signal m at time j

or, if the block is a dynamic one, then the block operator is

$D_{\text{k}} \left(S_{\text{q}}^{j}, S_{\text{r}}^{j} \right)$ = value of derivative of signal m at time j.

The variable S_{q}^{j} denotes that value of signal q at time j. The block operators are defined in Fig. A6.1-1. An auxiliary vector also stores information on dynamic blocks only. For the i-th occurring dynamic block having signal number m, the entry

$$L_{\text{dyn,i}} = m; \quad i = 1, ..., n_{\text{dyn}}$$

is created.

A unique signal number must also be assigned to each control signal. Recall a control signal is used to drive a plant variable and that the signal originates at the output of a block. If there are n_{ctl} control signals occurring in the user's input and the i-th one is assigned signal number m, then the following entries are created

$L_{\text{ctl,i}} = m; \ i = 1, \ldots, n_{\text{ctl}}.$

Further, if this control signal is taken from the output of block q, then the entry

$J_{\text{ctl,m}} = q.$

is also created.

The vectors L_{blk}, K_{blk}, $J1_{\text{blk}}$ and $J2_{\text{blk}}$ thus define the block diagram. Both the steady-state and transient solution methods access the elements in these vectors to march through the block diagram in a manner analogous to the step-wise progression up an axial mesh that is used in one-dimensional thermal and hydraulic analysis codes. In the control system problem, however, a logical relationship among blocks is substituted for the spatial relationship among fluid cells that occurs in thermal hydraulic problems.

6.3.2 Steady-State Solution

The set of equations given by Eq. (6.2-4) are a set of non-linear equations. Having the boundary conditions $u^*_{\text{mea}}(0)$ and $y^*_{\text{ctl}}(0)$, we must solve this set of equations for the initial conditions $u_{\text{dmd}}(0)$ and $x(0)$. However, before describing the solution method, we discuss two important considerations.

The first consideration is that the solution technique not require calculation of the inverse Jacobian of Eq. (6.2-4) with respect to the unknowns. Any non-linear equation solver that does will fail in certain cases when in fact a solution exists. If, for example, one of the unknowns feeds into a block that has a deadband region, then during iteration on the unknowns the input to this block may end up in the deadband zone. Then the derivative of the right-hand side of Eq. (6.2-4) will be zero with respect to the unknown, in which case the inverse of the Jacobian does not exist. The second consideration involves the relationship between the number of equations and the number of unknowns to be solved for. If the control system is properly designed then the equations are either square or they are over-determined. In the later case, the number of equations exceeds the number of unknowns. The solution method must be able to find a solution if it exists in either case. As for the non-linear equation solver used in this work, it will handle both the singular Jacobian and the over-determined equation situations. The solver is based on a least squares technique that is described in Ref. 6-3. When used in a case with deadband as described above, the equation solver will return a true solution if one exists. However, the returned solution may in fact be one of an infinite number.

The non-linear equation solver uses an iterative process to converge to the values of $u_{\text{dmd}}(0)$ and $x(0)$ that satisfy Eq. (6.2-4). The equation solver provides successively more refined estimates for u_{dmd} and $x(0)$ at the start of every iteration and expects the right-hand side of Eq. (6.2-4) to be calculated for each new set of values. Values for the right-hand side are generated by marching through the block diagram in the sequence defined by the vectors L_{blk}, K_{blk}, $J1_{\text{blk}}$ and $J2_{\text{blk}}$. The marching procedure is as follows. Suppose S^P_{m} denotes the value of signal m on the p-th iterate. Assume that if the signal S^P_{m} is either a measured signal or a control signal that it has been set with the respective boundary condition contained in $u_{\text{mea}}(0)$ or $y_{\text{ctl}}(0)$. Assume also that if the signal S^P_{m} corresponds to one of the unknowns in $u_{\text{dmd}}(0)$ or $x(0)$ that the equation solver has made its estimate for the signal prior to the p^{th} pass through the block diagram and the S^P_{m} has been set to this value. Then the value of the right hand-site of Eq. (6.2-4) for the p^{th} iteration is calculated using

$$
\begin{aligned}
&i = 1, \ldots n_{\text{blk}} \\
&m = L_{\text{blk,i}};\ k = K_{\text{blk,m}};\ q = J1_{\text{blk,m}};\ r = J2_{\text{blk,m}} \\
&S^P_{\text{m}} = 0,\ k = 4 \\
&S^P_{\text{m}} = F_{\text{k}}\left(S^P_{\text{q}},\ S^P_{\text{r}}\right),\ k = 1, 2, 3, 8, \ldots 21\ .
\end{aligned}
$$

On completion the elements of f are

$$
S^P_{\text{J1}_{\text{blk}},\text{L}_{\text{dyn}},\text{i}}, i = 1, \ldots n_{\text{dyn}}
$$

and the elements of g are

$$
S^P_{\text{J}_{\text{ctl}},\text{L}_{\text{ctl}},\text{i}}, i = 1, \ldots n_{\text{ctl}}\ .
$$

6.3.3 Transient Solution

The numerical techniques used to solve Eq. (6.2-3) are based on explicit differencing and a numerical marching procedure. The numerical techniques have performed well for those problems examined to date. A time step control mechanism automatically adjusts time step size to maintain a specified level of accuracy. This usually results in a step size smaller than the time constant of the fastest component which is often a sensor. When the equations are stiff, other solution techniques may offer a computationally more efficient solution. However, experience has shown that the order of the control equations is usually small and that the computational demands of the current scheme are reasonable.

The block diagram is advanced across a time step in two phases. In the first step, the block signals are updated to the start of the time step. This involves setting the measured and demand signals and then marching through the block diagram while holding dynamic block signals constant

$$
\begin{aligned}
& i = 1, \ldots n_{\text{blk}} \\
& m = L_{\text{blk,i}};\ k = K_{\text{blk,m}};\ q = J1_{\text{blk,m}};\ r = J2_{\text{blk,m}}\ , \\
& S_{\text{m}}^{j} = \frac{S_{\text{q}}^{j} - S_{\text{q}}^{j-1}}{t^{j} - t^{j-1}},\ k = 4\ , \\
& S_{\text{m}}^{j} = F_{\text{k}}\left(S_{\text{q}}^{j}, S_{r}^{j}\right),\ k = 1, 2, 3, 8, \ldots 21\ , \\
& S_{\text{m}}^{j} = S_{\text{m}}^{j},\ k = 5, 6, 7\ .
\end{aligned}
$$

Then in the second step, the block signals are advanced across the time step.

$$
\begin{aligned}
& i = 1, ...n_{\text{blk}} \\
& m = L_{\text{blk,i}};\ k = K_{\text{blk,m}};\ q = J1_{\text{blk,m}};\ r = J2_{\text{blk,m}}\ , \\
& S_{\text{m}}^{j+1} = \frac{S_{\text{q}}^{j} - S_{\text{q}}^{j-1}}{t^{j} - t^{j-1}},\ k = 4\ , \\
& S_{\text{m}}^{j+1} = F_{\text{k}}\left(S_{\text{q}}^{j+1}, S_{\text{r}}^{j+1}\right),\ k = 1, 2, 3, 8, \ldots 21 \\
& S_{\text{m}}^{j+1} = D_{\text{k}}\left(\frac{S_{\text{q}}^{j} + S_{\text{q}}^{j+1}}{2}, \frac{S_{\text{r}}^{j} + S_{\text{r}}^{j+1}}{2}\right)\left(t^{j+1} - t^{j}\right) + S_{\text{m}}^{j},\ k = 5, 6, 7\ .
\end{aligned}
$$

On completion the elements of x^{j+1} are stored in

$$S_{\text{L}_{\text{dyn,i}}}^{j+1} i = 1, \ldots n_{\text{dyn}}\ .$$

An accurate and stable solution to both the control equations and the plant equations is obtained by controlling the basic time step size known as a subinterval. The initial size of a new subinterval is obtained by SAS4A/SASSYS-1 by extrapolating rates of change in the plant from the previous subinterval. The control equations are advanced first over this new subinterval according to the algorithm just described. Two time step control mechanisms can come into effect during integration of the control equations.

The first time step mechanism attempts to limit the error in the control equation solution that results from numerically integrating over the subinterval. An initial estimate for this error is made after the integration algorithm has obtained a solution at the end of the current subinterval. The estimate is made for each element of the vector x (i.e. dynamic blocks) by first estimating a value at the end of the current subinterval by linearly extrapolating the change across the previous subinterval:

$$S_{\text{m,e}}^{j+1} = S_{\text{m}}^{j} + \frac{S_{\text{m}}^{j} - S_{\text{m}}^{j-1}}{t^{j} - t^{j-1}}\left(t^{j+1} - t^{j}\right)\ , \tag{6.3-1}$$

where

$$m = L_{\text{dyn,i}};\ i = 1, \ldots n_{\text{dyn}}\ . \tag{6.3-2}$$

If S_{m}^{j+1} is the value calculated by the integration algorithm then the error estimate is

$$e_{\text{m}}^{j} = \frac{\left|S_{\text{m}}^{j+1} - S_{\text{m,e}}^{j+1}\right|}{\left|S_{\text{m}}^{j}\right| + \text{F5SIG}\,(m)}$$

where F5SIG (m) is the zero crossing parameter supplied by the user as discussed below. The solution has converged if the quantity e_{m}^{j} is less than the user-supplied value for the error criterion EPSCS. If the solution

has not converged, then for purposes of control system integration only, the subinterval is bisected into two substeps and the control equations are again advanced over the subinterval. The error is again computed using Eq. (6.3-2) but using the value that resulted from the previous integration in place of $S_{\mathrm{m,e}}^{j+1}$. If the subinterval is still not converged, it is again bisected so now there are four substeps in the subinterval. This process is repeated until the error between successive iterations as defined by Eq. (6.3-2) is less than the input value for EPSCS.

The second time step mechanism limits the relative change in the control solution over a single subinterval. Large and unrestricted changes can lead to instability between the control system solution and the plant solution. After the subinterval has converged as described above, the relative change in control signals is computed via

$$C_{\mathrm{m}}^{j} = \frac{\left| S_{\mathrm{m}}^{j+1} - S_{\mathrm{m}}^{j} \right|}{\left| S_{\mathrm{m}}^{j} \right| + \mathrm{F5SIG}\,(m)} \tag{6.3-3}$$

where

$$m = L_{\mathrm{ctl,i}};\ i = 1, \ldots n_{\mathrm{ctl}}\ .$$

where F5SIG (m) is the zero crossing parameter whose value is supplied by the user. For the m that gives the largest value of C_{m}^{j} if this value C_{m}^{j} is greater than the user-supplied relative change criterion EPSCPL, then the subinterval time step is cutback so that the relative change EPSCPL is just met. The subinterval cutback size is obtained by linear interpolation so that the new size is the value of Δt that satisfies

$$\frac{\Delta t}{\mathrm{EPSCPL}} = \frac{t^{j+1} - t^{j}}{C_{\mathrm{m}}^{j}} \tag{6.3-4}$$

If the subinterval time step is cutback, then the control system integration starts over again using the new subinterval size.

A third time step mechanism is used to limit the relative change in the plant solution across a subinterval. This mechanism is analogous to the second time step mechanism and is described in Ref. 6-1.

6.4 Guide to User Application

This section provides some guidelines that should help tie together the model equations and solution techniques just described and the card input description given in Section 6.8. In this section signal definition rules that must be observed are stated, model capabilities are highlighted, and rules of thumb for choosing the values of solution control parameters are given.

6.4.1 Signal Definition Rules

Signals are defined through the input deck and the definitions must conform to certain rules. Any of the four signal kinds can appear anywhere in the signal card region of the input deck, subject to the following rules.

1. A block signal that is output from other than an integrator, lag compensator or lead-lag compensator must have been previously defined in the input stream before it can be used as an input to another block. This rule is intended to avoid circular references and to maintain proper sequencing of signals during numerical integration.
2. A demand signal or measured signal must pass through at least one block before it can be used as a control signal.
3. Each signal must be assigned a unique signal number between 1 and 998.

The card format for defining a signal is given in Section 6.8.

6.4.2 Units

Generally all measured signals are in MKS units while all control signals should be calculated in these same units. The exceptions are those signals that are normalized to a steady state value; these are appropriately noted in Table 6.8.3.

The convention for demand signals is that demand tables are always entered by the user, normalized to a time zero value of unity. The actual value for a demand signal is calculated in the code by multiplying the current time entry in the demand table by the initial condition value. The next subsection describes how the initial condition value is obtained.

The units of a block output signal are determined solely by the units of the input signals and any conversion factors that are entered by the user as constants on the block definition card.

6.4.3 Initial Conditions

In order to begin a transient calculation, initial condition values are required for demand signals and for the integrator, lag compensator and lead-lag compensator blocks. There are basically three options available for setting these values. In the first option, all values are supplied by the user through input cards: in this case the steady state solution finder is bypassed. If the user is seeking the steady state solution, then a null transient may have to be run. In the second option, the steady state solution finder is used to solve for the steady state values. In the final option, a mixed set of initial conditions are used with some values read directly from the signal cards while the remaining values are solved for such that Eq. (6.2-4) is satisfied. The card input data required for each of these options is described below.

If the initial condition values are to be read from the input cards then the steady state solution finder should be bypassed by setting the J1SIG field on the 999 card to '0'. Then the value for a demand signal and for the block signal of each integrator, lag compensator and lead-lag compensator is taken from the F4SIG field on the associated signal definition card.

If the initial condition values are to be calculated by the steady state solution finder then the J1SIG field on the 999 card is set to '1'. An initial guess for each demand signal and integrator initial condition variable must be supplied on the F4SIG field of the signal definition card. In addition the F3SIG field must be set to '0'. As a rule of thumb, the initial guess should be within 15% of the actual steady state value to ensure convergence. The lag compensator and lead-lag compensator are special cases and do not require initial condition information from the user.

Finally, if a mixed set of initial conditions is to be used, the card data is identical to the case directly above, except for those demand and integrator signals whose initial conditions are to be read from cards. For these signals the F3SIG field is set to '1.0' and the F4SIG field is set to the initial condition value desired.

6.4.4 Solution Accuracy

The control system modeling capability attempts to limit the solution error that is introduced during the numerical integration of the control equations over a subinterval. Recall the error is controlled by repeatedly bisecting the subinterval time step into substeps until integrating across the subinterval gives a relative error between successive iterations that is less than the user-supplied value for EPSCS. [The method was described in Section 6.3]. The value of EPSCS is input on a table card and occupies location 8001. A value of 0.01 is suggested for most applications.

6.4.5 Solution Stability

The modeling capability also attempts to maintain a stable solution to the coupled control system and plant equations. The basic idea is that stability is enhanced if the relative change in a control signal across a subinterval is maintained less than the user supplied value for EPSCPL. [The method was described in Section 6.3]. The value of EPSCPL is input on a table card and occupies location 8002. A value of 0.1 is suggested for most applications.

6.4.6 Zero Crossing Parameter

The zero crossing parameter in Eq. (6.3-2) is intended to prevent unnecessarily small time step size when a signal passes close to zero. The situation we seek to avoid occurs when the zero crossing parameter F5SIG is zero. Then the denominator in Eq. (6.3-2) is very small so that the relative error is very large. Time step size is severely reduced even though the absolute error in the signal may well be acceptably small. The solution is to control absolute error at the zero crossing and we do it through the relative error control mechanisms associated with Eq. (6.3-2) by proper choice of a value for F5SIG.

The appropriate value of F5SIG is problem dependent and is selected by the user for input to the code. The goal is to select a value that gives a desired level of absolute error near the zero crossing yet does not significantly impact the calculation of relative error away from the zero crossing. To do so we note that the code controls integration error using Eq. (6.3-2) so that on convergence the solution satisfies

$$\left|S_{\mathrm{m}}^{j+1} - S_{\mathrm{m,e}}^{j+1}\right| = \mathrm{EPSCS}\left(\left|S_{\mathrm{m}}^{j}\right| + \mathrm{F5SIG}\,(m)\right) \tag{6.4-1}$$

where the value of m is restricted to those signals that are output by dynamic blocks. Near the zero crossing S_{m}^{j} will be insignificant so that Eq. (6.4-1) is equivalently

$$\mathrm{F5SIG}\,(m) = \frac{\left|S_{\mathrm{m}}^{j+1} - S_{\mathrm{m,e}}^{j+1}\right|}{\mathrm{EPSCS}}\,. \tag{6.4-2}$$

Note that the numerator is the absolute error in the solution at convergence. We can arrange for the numerator to take on a specific value by appropriately choosing the value of F5SIG (m) once the value of EPSCS has been selected. For example, suppose we want the absolute error on convergence near the zero crossing to be $S_{\mathrm{m}}^{o} \times 10^{-4}$ where S_{m}^{o} is the maximum magnitude signal m is to take on over all time. If, the sake of illustration, a value of 10^{-2} was input for EPSCS, then we can achieve our absolute error objective by calculating the value of F5SIG (m) from Eq. (6.4-2),

$$\mathrm{F5SIG}\,(m) = \frac{S_{\mathrm{m}}^{o} 10^{-4}}{10^{-2}} = S_{\mathrm{m}}^{o} 10^{-2}$$

Away from the zero crossing, the impact of F5SIG (m) is insignificant.

Similarly, the value of F5SIG (m) associated with a control signal should be selected as follows. The time step is adjusted down if necessary so that the largest relative change in a control signal is limited by Eq. (6.3-3) to

$$\left|S_{\mathrm{m}}^{j+1} - S_{\mathrm{m,e}}^{j}\right| = \mathrm{EPSCPL}\left(\left|S_{\mathrm{m}}^{j}\right| + \mathrm{F5SIG}\,(m)\right) \tag{6.4-3}$$

where the value of m is restricted to those signals that are control signals. Near the zero crossing S_{m}^{j}will be insignificant so that Eq. (6.4-3) is equivalently

$$\mathrm{F5SIG}\,(m) = \frac{\left|S_{\mathrm{m}}^{j+1} - S_{\mathrm{m}}^{j}\right|}{\mathrm{EPSCPL}}\,. \tag{6.4-4}$$

Note that the numerator is the absolute change in the solution across the time step. We can arrange for the numerator to take on a specific value by appropriately choosing the value of F5SIG (m) once the value of

EPSCPL has been selected. For example, suppose we want the absolute change in the control signal near the zero crossing to be as large as $S^o_{\mathrm{m}} \times 10^{-3}$ before time step size is reduced. If, for the sake of illustration, a value of 10^{-1} was input for EPSCPL, then the absolute change objective will be met if F5SIG (m) is calculated from Eq. (6.4-4),

$$\text{F5SIG}\,(m) = \frac{S^o_{\mathrm{m}} 10^{-3}}{10^{-1}} = S^o_{\mathrm{m}} 10^{-2}$$

Away from the zero crossing, the impact of F5SIG (m) on the control of fractional change is insignificant.

6.5 Control System Signals

One or more sequences of user-defined **signals** are used to define a control-system block diagram. The four types of signals are *measured*, *demand*, *block*, and *control*. Each is described in the sections that follow. Detailed information on the input requirements is provided in the *Appendix*.

6.5.1 Measured Signals

Measured signals represent the value of plant-state variables, such as pressure, temperature, or flow rate. Measured signals are implemented in SAS as a virtual data acquisition system (DAS). Unless otherwise noted, measured signal values are obtained at the beginning of a PRIMAR time step. These values are used to update the control system state in effect during the time step. That state is then reported along with other simulation output at the end of the time step.

6.5.2 Demand Signals

Demand signals represent the value of an external, time-dependent load. Demand signals are interpolated from a table of user-defined (time,value) pairs, where the values are assumed to be normalized to an initial-condition value.

6.5.3 Block Signals

Block signals represent mathematical or logical operations that define the control system's behavior. Input to block signals comes from measured, demand, or other block signals.

6.5.3.1 Variable Lag Compensator

Note: New in Version 5.3.

Theory

A *variable lag compensator* is similar to the lag compensator except that the delay time parameter, τ, can vary during the Control System simulation. An equation that expresses this behavior is

$$y(t) + \tau(t)\frac{dy}{dt} = u(t)$$

where

$u(t)$ = input driving function

$\tau(t)$ = delay (lag)

$y(t)$ = output function result

Assuming that the lag time is always positive, a solution to the above equation is

$$y(t) = y_0 e^{-\int_0^t \frac{dt'}{\tau(t)}} + e^{-\int_{t_0}^t \frac{dt'}{\tau(t)}} \int_0^t \frac{u(t')}{\tau(t')} e^{\int_{t_0}^{t'} \frac{dt''}{\tau(t'')}} dt'$$

where $y_0 = y(0)$ and t_0 is an arbitrary integration constant.

The complexity of the above solution can be simplified if we assume the input driving function and delay function are linear functions of time: $u(t) = u_0 + u't$ and $\tau(t) = \tau_0 + \tau' t$, respectively. With this assumption, the solution becomes

$$y(t) = y_0\left(1 + \frac{\tau'}{\tau_0}t\right)^{-\frac{1}{\tau'}} + u_0\left[1 - \left(1 + \frac{\tau'}{\tau_0}t\right)^{-\frac{1}{\tau'}}\right] + \frac{u'}{1+\tau'}\left[t - \tau_0\left[1 - \left(1 + \frac{\tau'}{\tau_0}t\right)^{-\frac{1}{\tau'}}\right]\right]$$

If τ is slowly varying with time and we take the limit that $\tau' \to 0$, we arrive at the solution for a constant τ during the time interval:

$$y(t) = y_0 e^{-\frac{t}{\tau_0}} + u_0\left(1 - e^{-\frac{t}{\tau_0}}\right) + u'\left[t - \tau_0\left(1 - e^{-\frac{t}{\tau_0}}\right)\right]$$

Implementation

The analytic solution for the variable lag compensator can be used to solve for the value of $y_1 = y(t_0 + \Delta t)$ at the end of a Control System sub-interval (time step) given the initial value $y_0 = y(t_0)$. Knowing the lag time at the beginning of the time step, τ_0, and the input driving function at the beginning and end of the time step, $u_0 = u(t_0)$ and $u_1 = u(t_0 + \Delta t)$, respectively, then the solution for y_1 at the end of the time step is

$$y_1 = y_0 e^{-\delta} + u_0\left(1 - e^{-\delta}\right) + (u_1 - u_0)\left[1 - \frac{1 - e^{-\delta}}{\delta}\right]$$

where $\delta = \frac{\Delta t}{\tau_0}$.

Because this solution is prone to numerical round-off errors, expansions for the exponential function are used:

$$e^{-\delta} = 1 - \delta + \frac{\delta^2}{2} - \frac{\delta^3}{6} + O\left[\delta^4\right]$$

and

$$\frac{1 - e^{-\delta}}{\delta} = 1 - \frac{\delta}{2} + \frac{\delta^2}{6} - \frac{\delta^3}{24} + O\left[\delta^4\right]$$

Using these expansions, the solution is written as

$$y_1 = y_0 \left[1 - \delta + \frac{\delta^2}{2}\right] + u_0 \left[\frac{\delta}{2} - \frac{\delta^2}{3}\right] + u_1 \left[\frac{\delta}{2} - \frac{\delta^2}{6}\right] + O\left[\delta^3\right]$$

To minimize numerical round-off errors in the code, the following form is used:

$$y_1 \approx y_0 \left[1 - \delta\left(1 - \delta\left(\frac{1}{2}\right)\right)\right] + u_0 \left[\delta\left(\frac{1}{2} - \delta\left(\frac{1}{3}\right)\right)\right] + u_1 \left[\delta\left(\frac{1}{2} - \delta\left(\frac{1}{6}\right)\right)\right]$$

which is second-order accurate in δ. This form also avoids the need to make expensive exponential function evaluations.

Use

A *variable lag compensator* is identified in the Control System input as signal type ITYPE = 23. It requires two input signals to define the time-dependent functions $u\,(t)$ and $\tau\,(t)$. These are identified by the input parameters J1 and J2, respectively. Input for the variable lag compensator is summarized in the table below.

Input	Description
ISIG	User-defined signal number
ITYPE = 23	Variable Lag Compensator type
J1	Signal # of driving function, $u(t)$
J2	Signal # of delay time, $\tau\,(t)$
F1	Multiplier for driving function [default: 1]
F2	(unused)
F3	(unused)
F4	Steady-state initial condition, $y\,(0)$ [default: $y\,(0) = u\,(0)$]
F5	Zero-crossing parameter (F5 > 0)

The input includes two optional parameters, F1 and F4, and one required parameter, F5. If the optional parameters are blank or zero, defaults will be applied as indicated in the table. For example, if F1 is non-zero, then the actual driving function will include a multiplier:

$$u\,(t) = \text{F1} \times \text{sig}\,(\text{J1})$$

Likewise, if F4 is non-zero, it is used to define the initial steady-state condition as $y\,(0) =$ F4. Otherwise, the steady-state solution to the equation is $y\,(0) = u\,(0)$.

The zero-crossing parameter, F5, is used in the same manner as with other Control System signals to stabilize convergence testing when the solution for $y\,(t)$ is near zero. The normalized error is approximated as follows to avoid divide-by-zero errors:

$$\text{error} = \left|\frac{\Delta y}{y}\right| \cong \frac{|\Delta y|}{|y| + \text{F5}}$$

Therefore, users can select a value for F5 that is small enough to ensure convergence when $y\,(t)$ is small, but large enough to avoid excessive iterations when small values for $y\,(t)$ do not have a significant impact on the overall solution.

Like the lag compensator, the variable lag compensator shifts both the phase and amplitude of the input signal $u\,(t)$. However, the phase and amplitude vary as a function of time through the additional input for $\tau\,(t)$. To

demonstrate this, consider the following conditions:

$$u(t) = \sin\left(\frac{2\pi}{5}t\right)$$
$$\tau(t) = 1.9\sin\left(\frac{2\pi}{50}t\right) + 2$$
$$y(0) = 2$$

These can be represented in the Control System input as follows:

```
INCONT     5

# Time: t
1 -55 0 0 0.0 0.0 0.0 0.0 0.0

# Driving Signal: Sin(2πt/5)
11 22 1 0 1.0 1.2566370 0.0 0.0

# Lag Time Variation: Sin(2πt/50)
12 22 1 0 1.0 .12566370 0.0 0.0

# Constant: 2
13 21 0 0 2.0 0.0 0.0 0.0 0.0

# Lag Time: 1.9Sin(2πt/50) + 2
17 1 12 13 1.9 1.0 1.0 0.0 0.0

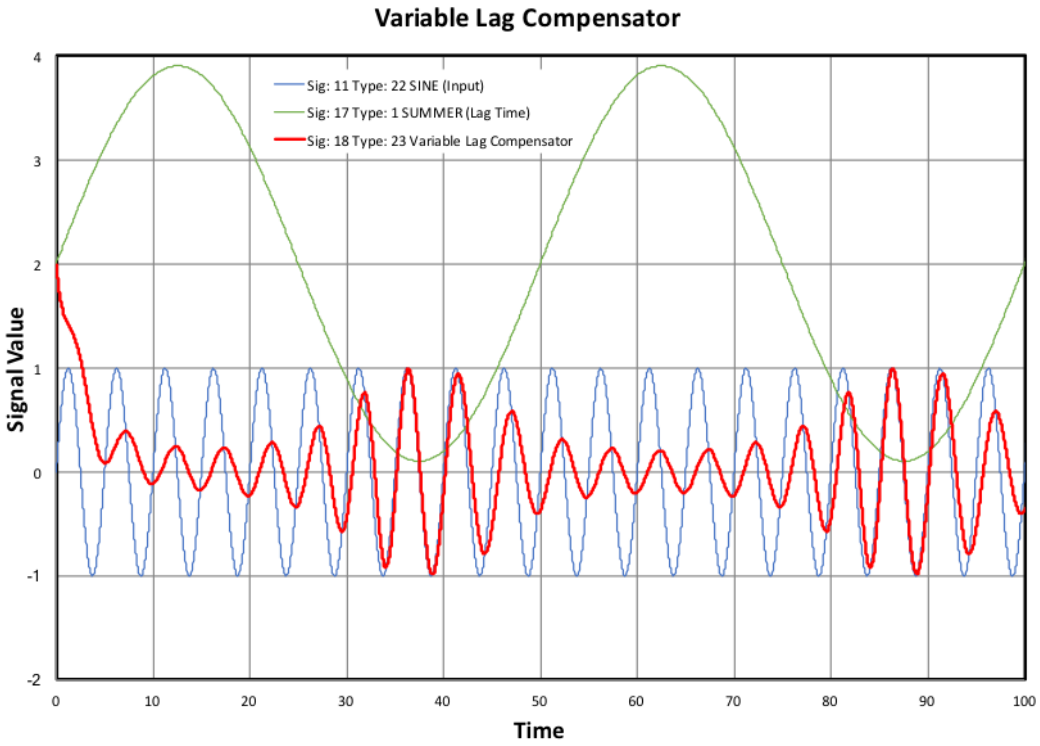
# Variable Lag Compensator(11,17)
# ITYPE J1 J2 F1 F4 F5
18 23 11 17 1.0 0.0 0.0 2.0 0.001

999 1

8001 2 0.00000001 0.00000001

END
```

Using the above Listing in a SAS4A/SASSYS-1 input file would produce the following Control System output:



Note that because F4 = 2, the variable lag compensator starts at the non-equilibrium value $y\left(0\right) = 2$. As the lag time $\tau\left(t\right)$ rises and falls, the delay in the input signal becomes longer and shorter, respectively. Similarly, the amplitude of the output decreases and increases.

6.5.4 Control Signals

Control signals are used to set the value of a SAS variable during the simulation. Currently, a limited number of simulation variables can be controlled. The primary purpose of control signals is to affect reactivity (e.g. scram), pump torque, and valve coefficients.

6.5.4.1 Simulation Trip

The simulation trip control signal (JTYPE=-10) is used to stop the SAS simulation based on the value of an input signal. When the input signal changes from false (a value of zero) to true (a non-zero value), code execution will end at the end of the main time step. Since the input signal is treated as a logical, a large zero crossing parameter (e_z, entered in F6SIG) is recommended to avoid control system cutbacks. A value of 1E+09 is suggested.

The control system trips the simulation by setting the maximum number of time steps to the current main time step number when the input signal triggers the trip. To restart from a case that was ended by the simulation trip control signal, the user should reset **MAXSTP** in the restart case input file to the desired value so that the simulation can continue.

A simulation trip control signal can be configured as unlatched (J2SIG = 0) or latched (J2SIG = 1). An unlatched simulation trip control signal is capable of tripping the simulation again after a restart, while a latched simulation trip control signal cannot.

6.6 References

6-1. R. B. Vilim, Unpublished information, Argonne National Laboratory, 1986.

6-2. R. B. Vilim, T. Y. C. Wei, and F. E. Dunn, "Generalized Control System Modeling for Liquid-Metal Reactors," *Nucl. Sci. Eng.*, vol. 99, pp. 183-196, July, 1988.

6-3. J. J. More, "The Levenberg-Marquardt Algorithm: Implementation and Theory," *Proceedings of the Biennial Conference on Numerical Analysis*, Dundee, Scotland, June 28 - July 1, 1977, Springer-Verlag, New York, pp. 105-116, 1978.

6.7 Nomenclature

Symbol	Definition	Units
C	Normalized signal change	
D	Dynamic block operator	
e	Error estimate	
F	Static block operator	
i	Block index	
J	Block input signal	
j	Time step index	
K	Block type index	
k	Block type index	
L	Block output signal number	
m	Block output signal number	
q	Block input signal	
r	Block input signal	
S	Signal value	
t	Time	s
u	Input variable	
x	State variable	
y	Output variable	

6.8 Appendix 6.1: Control System Input Description

This appendix contains a description of the SAS4A/SASSYS-1 input block assigned to the control system model.

6.8.1 Input Block Structure

The input block structure is identical to the standard SAS4A/SASSYS-1 input block structure in all but one respect. A new card format known as a signal card has been introduced. These cards immediately follow the block identifier card and precede the standard data cards. The ordering of the different card types is depicted in the diagram below.

```
block identifier card
signal card # 1
signal card # 2

.
.
.

signal card # n
end of signal card

data card # 1
data card # 2

.
.
.

data card # m
block delimiter card
```

6.8.2 Block Identifier Card

Block identifier cards are described in Section 2.7. For the control system input block the number is 5 and the block name is INCONT.

6.8.3 Signal Cards

```
form = "(4I5, 5F10.0)"
read(*,form) ISIG, JTYPE, J1SIG, J2SIG, F1SIG, F2SIG, F3SIG, F4SIG, F5SIG
```

A signal card is used to define a signal in the user's control block diagram. There are four signal types: measured, demand, block and control. Each signal must be assigned a unique signal identification number using the ISIG field. The value of ISIG must lie between 1 and 998, or between 1000 and 10000. The remaining variables that are used to define control system signals are defined in Table 6.8.1.

Table 6.8.1: Signal Card Format

Column	Fortran Symbol	Definition	Variable Type
1	ISIG	Signal number	Integer
6	JTYPE	Signal type	Integer
11	J1SIG	Signal descriptor 1	Integer
16	J2SIG	Signal descriptor 2	Integer
21	F1SIG	Constant 1	Real
31	F2SIG	Constant 2	Real
41	F3SIG	Constant 3	Real
51	F4SIG	Constant 4	Real
61	F5SIG	Constant 5	Real

6.8.3.1 Measured Signal

A measured signal makes available to the block diagram the present value of a referenced SAS4A/SASSYS-1 variable. The correspondence between the variable that is referenced and the signal card data field values is given in Table 6.8.3. Note that all measured signals have a JTYPE value between -50 and -89 or between -101 and -136.

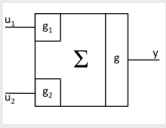
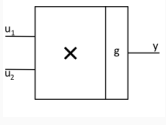
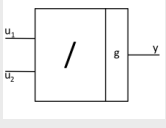
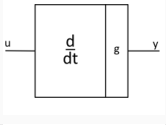
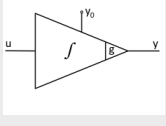
6.8.3.2 Demand Signal

A demand signal makes available to the block diagram the product of the current value of a time dependent function defined by the user through a demand table and an initial condition value. A demand table is a set of (`CTLTAB`, `CTLTIM`) ordered pairs supplied by the user in the format of Table 6.8.4. The code obtains the demand signal value by linearly interpolating among the table entries using the current time. The initial value is obtained as described in Section 6.4. The correspondence between the demand table and the signal card data fields is given in Table 6.8.3. Note that a *demand signal* has a JTYPE value of -90.

6.8.3.3 Block Signal

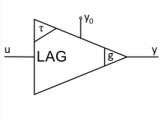
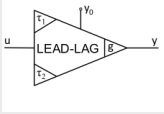
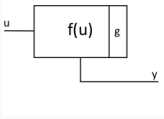
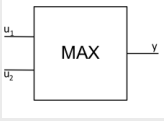
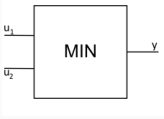
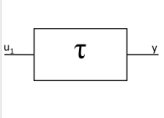
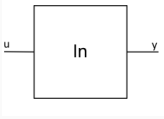
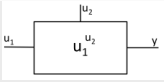
A block signal makes available to the block diagram the value at the output of a block. The correspondence between the block characteristics and the signal card data fields is given in Table 6.8.3. Note that all block signals have a JTYPE value between 1 and 23. A measured, demand or block signal can be used as an input to a block by specifying on the block's signal definition card the signal identification number assigned to the input signal. The signals input to each block type are combined according to the mathematical expressions given in Table 6.8.2.

Table 6.8.2: Summary of Mathematical Block Signals

JTYPE	Block	Mathematical Form	Representation	Type
1	Summer	$y = g(g_1 u_1 + g_2 u_2)$		function
2	Multiplier	$y = g u_1 u_2$		function
3	Divider	$y = g \frac{u_1}{u_2}$		function
4	Differential	$y = g \frac{d}{dt} u$		function
5	Integrator	$y = y_0 + g \int_0^t u \, dt$		dynamic

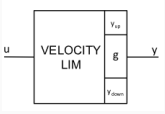
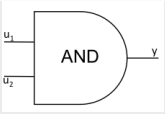
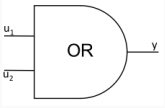
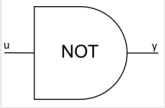
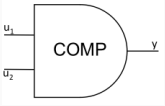
continues on next page

Table 6.8.2 – continued from previous page

JTYPE	Block	Mathematical Form	Representation	Type
6	Lag Compensator (See also *Variable Lag Compensator*)	$y + \tau \frac{d}{dt} y = g\, u(t)$ $y\,(0) = y_0$		dynamic
7	Lead-Lag Compensator	$y + \tau_1 \frac{dy}{dt} = g\left(u + \tau_2 \frac{du}{dt}\right)$ $y(0) = y_0$		dynamic
8	Function Generator	$y = g\, f(u)$		table
9	Maximum Value	$y = \max\,(u_1, u_2)$		function
10	Minimum Value	$y = \min\,(u_1, u_2)$		function
11	Time Delay	$y = u(0); 0 \le t \le T$ $y = u(t - \tau); t > T$		function
12	Natural Logarithm	$y = \ln u$		function
13	Exponentiation	$y = u_1^{u_2}$		function

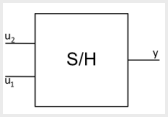
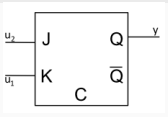
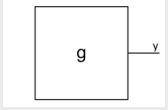
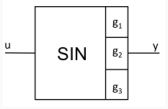
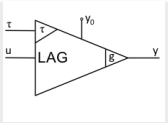
continues on next page

Table 6.8.2 – continued from previous page

JTYPE	Block	Mathematical Form	Representation	Type
14	Velocity limiter	$y = y_{\text{down}}; gu < y_{\text{down}}$ $y = y_{\text{up}}; gu > y_{\text{up}}$ $y = gu$; otherwise $y_{\text{down}} = y\,(t-h) - h\,v_{\text{down}}$ $y_{\text{up}} = y\,(t-h) + h\,v_{\text{up}}$		function
15	AND	$y = 1; u_1 > 0, u_2 > 0$ $y = 0$; otherwise		logic
16	OR	$y = 0; u_1 \leq 0, u_2 \leq 0$ $y = 1$; otherwise		logic
17	NOT	$y = 1 \;\; u \leq 0$ $y = 0 \;\; u > 0$		logic
18	Comparator	$y = 0$ if $u_1 < u_2$ $y = 1$ if $u_1 \geq u_2$		logic

continues on next page

Table 6.8.2 – continued from previous page

JTYPE	Block	Mathematical Form	Representation	Type
19	Sample and Hold	$y(t) = u_2(t)$ when $u_1(t) \leq 0$ $y(t) = u_2(t_s)$ when $u_1(t) \geq 0; t_s < t$ where $u_1(t') \leq 0; t_- \leq t' < t_s$		function
20	J-K Flip-Flop	$y^{n+1} = Q^n; u_1 \leq 0, u_2 \leq 0$ $y^{n+1} = 0; u_1 > 0, u_2 \leq 0$ $y^{n+1} = 1; u_1 \leq 0, u_2 > 0$ $y^{n+1} = \overline{Q}^n; u_1 > 0, u_2 > 0$		logic
21	Constant	$y = g$		function
22	Sine	$y = g_1 \sin(g_2 u(t) + g_3)$		function
23	Variable Lag Compensator	$y + \tau(t)\frac{d}{dt}y = g\,u(t)$ $y(0) = y_0$		dynamic

6.8.3.4 Control Signals

A control signal is used to set the value of a SAS4A/SASSYS-1 variable equal to the value of a block signal. The correspondence between the block signal and the SAS4A/SASSYS-1 variable and the signal card data fields is given in Table 6.8.3. Note that all control signals have a JTYPE value between -1 and -10.

6.8.3.5 End of Signals

A sequence of signal definition cards is delimited by a signal card with the ISIG field entry equal to '999'.

This card also contains flags for the binary output file print interval and control of the stead state solution finder. First, the absolute value of the JTYPE field for the 999 card is sets the print interval for control system results output to the binary output file CONTROL.dat. Second, the J1SIG field is used to determine whether the steady state solution finder is to be used. An entry of '1' indicates that the steady state solution finder is to be used, while any other entry in this field causes the solution finder to be bypassed. (A discussion of the initial condition option is given in Section 6.4). Finally, the J2SIG field allows the user to control the amount of steady state output generated. An entry of '1' produces an extended output for trouble shooting purposes, while any other entry produces a standard output.

The JTYPE field is also used to generate an extended printout during the transient for debug purposes. The debug is generated by setting the JTYPE field of the 999 card to a negative value. The printout begins at the time specified on the F1SIG field.

Table 6.8.3: Signal Cards

Type	Variable	JTYPE	J1SIG	J2SIG	F1SIG	F2SIG	F3SIG	F4SIG	F5SIG
Measured	Compressible volume pressure, PRESL3	-50	ICV[1]					y_0 (opt)	
Measured	Liquid segment flowrate; FLOSL3	-51	ISGL[2]					y_0 (opt)	
Measured	Liquid cover gas interface elevation, ZINTR3	-52	ICV					y_0 (opt)	
Measured	Liquid mass, XLQMS3	-53	ICV					y_0 (opt)	
Measured	Cover gas volume, VOLGC3	-54	ICV					y_0 (opt)	
Measured	Time	-55						y_0 (opt)	
Measured	Pump head, HEADP3	-56	IPMP[3]					y_0 (opt)	
Measured	Liquid temperature, TLQCV3	-57	ICV					y_0 (opt)	
Measured	Liquid density, DNSCV3	-58	ICV					y_0 (opt)	
Measured	Wall temperature, TWLCV3	-59	ICV					y_0 (opt)	
Measured	Cover gas pressure, PRESG3	-60	ICV					y_0 (opt)	

continues on next page

Table 6.8.3 – continued from previous page

Type	Variable	JTYPE	J1SIG	J2SIG	F1SIG	F2SIG	F3SIG	F4SIG	F5SIG
Measured	Cover gas mass, GASMS3	-61	ICV					y_0 (opt)	
Measured	Cover gas temperature, TGASC3	-62	ICV					y_0 (opt)	
Measured	Not used	-63						y_0 (opt)	
Measured	Liquid segment temperature, TSLIN3	-64	ISGL	Inlet=1 Outlet=2				y_0 (opt)	
Measured	Pump speed, PSPED3	-65	IPMP					y_0 (opt)	
Measured	Core channel coolant flowrate, CHFL03	-66	ICH[4]	Inlet=1 Outlet=2				y_0 (opt)	
Measured	Liquid node temperature, TLNOD3	-67	Node number, INOD					y_0 (opt)	
Measured	Wall node temperature, TWNOD3	-68	Node number, INOD					y_0 (opt)	
Measured	Liquid element temperature, TELEM	-69	Element number, IEL	Inlet=1 Outlet=2				y_0 (opt)	
Measured	Not used	-70						y_0 (opt)	
Measured	Core channel outlet temperature, CHFCOF	-71	ICH	Inlet=1 Oulet=2				y_0 (opt)	
Measured	Normalized reactor power, DEXP (POWVA (3,1))	-72						y_0 (opt)	
Measured	Normalized fission power, POWFSO × AMPO	-73						y_0 (opt)	
Measured	Normalized decay heat, SUM(POWWT(i) × POWDKH(i))	-74						y_0 (opt)	

continues on next page

Table 6.8.3 – continued from previous page

Type	Variable	JTYPE	J1SIG	J2SIG	F1SIG	F2SIG	F3SIG	F4SIG	F5SIG
Measured	Equivalent Circuit EM Pump Voltage	-75	IPMP					y_0 (opt)	
Measured	Equivalent Circuit EM Pump Frequency	-76	IPMP					y_0 (opt)	
Measured	Equivalent Circuit EM Pump Current	-77	IPMP					y_0 (opt)	
Measured	Equivalent Circuit EM Pump Phase Angle	-78	IPMP					y_0 (opt)	
Measured	Not used	-79,… -82						y_0 (opt)	
Measured	Steam generator, feed-water mass flowrate in	-83		SG number				y_0 (opt)	
Measured	Steam generator, feed-water enthalphy in	-84		SG number				y_0 (opt)	
Measured	Steam generator, steam mass flowrate	-85		SG number				y_0 (opt)	
Measured	Steam generator, steam temperature out	-86		SG number				y_0 (opt)	
Measured	Steam generator, steam pressure	-87		SG number				y_0 (opt)	
Measured	Steam generator, water level	-88		SG number				y_0 (opt)	
Measured	Steam generator, steam enthalpy out	-89		SG number			Initial condition flag	y_0	
Demand	Demand Table	-90	Demand table number	Number of entries in table				y_0	
Measured	Fuel Centerline Temperature	-101	ICH	MZ[5]	scale[6] (opt)	offset[7] (opt)		y_0 (opt)	
Measured	Fuel Average Temperature	-102	ICH	MZ	scale (opt)	offset (opt)		y_0 (opt)	
Measured	Fuel Surface Temperature	-103	ICH	MZ	scale (opt)	offset (opt)		y_0 (opt)	
Measured	Clad Inner Wall Temperature	-104	ICH	MZ	scale (opt)	offset (opt)		y_0 (opt)	

continues on next page

Table 6.8.3 – continued from previous page

Type	Variable	JTYPE	J1SIG	J2SIG	F1SIG	F2SIG	F3SIG	F4SIG	F5SIG
Measured	Clad Mid Wall Temperature	-105	ICH	MZ	scale (opt)	offset (opt)		y_0 (opt)	
Measured	Clad Outer Wall Temperature	-106	ICH	MZ	scale (opt)	offset (opt)		y_0 (opt)	
Measured	Coolant Temperature	-107	ICH	MZC[8]	scale (opt)	offset (opt)		y_0 (opt)	
Measured	Coolant Pressure	-108	ICH	MZC	scale (opt)	offset (opt)		y_0 (opt)	
Measured	Coolant Saturation Temperature	-109	ICH	MZC	scale (opt)	offset (opt)		y_0 (opt)	
Measured	Coolant Boiling Margin	-110	ICH	MZC	scale (opt)	offset (opt)		y_0 (opt)	
Measured	Coolant Average Temperature	-111	ICH	MZC	scale (opt)	offset (opt)		y_0 (opt)	
Measured	Structure Inner Temperature	-112	ICH	MZC	scale (opt)	offset (opt)		y_0 (opt)	
Measured	Structure Outer Temperature	-113	ICH	MZC	scale (opt)	offset (opt)		y_0 (opt)	
Measured	Reflector Inner Temperature	-114	ICH	MZC	scale (opt)	offset (opt)		y_0 (opt)	
Measured	Reflector Outer Temperature	-115	ICH	MZC	scale (opt)	offset (opt)		y_0 (opt)	
Measured	Peak Fuel Temperature	-116	ICH	ICH2[9] (opt)	scale (opt)	offset (opt)		y_0 (opt)	
Measured	Peak Clad Temperature	-117	ICH	ICH2 (opt)	scale (opt)	offset (opt)		y_0 (opt)	
Measured	Peak Coolant Temperature	-118	ICH	ICH2 (opt)	scale (opt)	offset (opt)		y_0 (opt)	
Measured	Minimum Boiling Margin	-119	ICH	ICH2 (opt)	scale (opt)	offset (opt)		y_0 (opt)	
Measured	Coolant Inlet Temperature	-120	ICH		scale (opt)	offset (opt)		y_0 (opt)	
Measured	Coolant Inlet Pressure	-121	ICH		scale (opt)	offset (opt)		y_0 (opt)	
Measured	Coolant Inlet Flowrate	-122	ICH		scale (opt)	offset (opt)		y_0 (opt)	
Measured	Coolant Outlet Temperature	-123	ICH		scale (opt)	offset (opt)		y_0 (opt)	
Measured	Coolant Outlet Pressure	-124	ICH		scale (opt)	offset (opt)		y_0 (opt)	
Measured	Coolant Outlet Flowrate	-125	ICH		scale (opt)	offset (opt)		y_0 (opt)	
Measured	Maximum Coolant Outlet Temperature	-126			scale (opt)	offset (opt)		y_0 (opt)	
Measured	Minimum Coolant Outlet Temperature	-127			scale (opt)	offset (opt)		y_0 (opt)	

continues on next page

Table 6.8.3 – continued from previous page

Type	Variable	JTYPE	J1SIG	J2SIG	F1SIG	F2SIG	F3SIG	F4SIG	F5SIG
Measured	Pin Bundle ΔT	-128	ICH		scale (opt)	offset (opt)		y_0 (opt)	
Measured	Pin Bundle ΔP	-129	ICH		scale (opt)	offset (opt)		y_0 (opt)	
Measured	Assembly Bundle ΔT	-130	ICH		scale (opt)	offset (opt)		y_0 (opt)	
Measured	Assembly Bundle ΔP	-131	ICH		scale (opt)	offset (opt)		y_0 (opt)	
Measured	Assembly Power	-132	ICH		scale (opt)	offset (opt)		y_0 (opt)	
Measured	Pin Linear Power	-133	ICH	MZ	scale (opt)	offset (opt)		y_0 (opt)	
Measured	Peak Pin Linear Power	-134	ICH		scale (opt)	offset (opt)		y_0 (opt)	
Measured	Fission Gas Plenum Temperature	-135	ICH		scale (opt)	offset (opt)		y_0 (opt)	
Measured	Fission Gas Plenum Pressure	-136	ICH		scale (opt)	offset (opt)		y_0 (opt)	
Block	Summer	1	ISIG1[10]	ISIG2[11]	g_1	g_2	g	y_0 (opt)	
Block	Multiplier	2	ISIG1	ISIG2	g			y_0 (opt)	
Block	Divider	3	ISIG1	ISIG2	g			y_0 (opt)	
Block	Differentiator	4	ISIG1		g			y_0 (opt)	
Block	Integrator	5	ISIG1		g		Initial condition flag	y_0	e_z[12]
Block	Lag compensator	6	ISIG1		g	τ		y_0[a]	e_z
Block	Lead-lag compensator	7	ISIG1		g	τ_1	τ_2	y_0[a]	e_z
Block	Function generator	8	ISIG1	Function generator table number	g			y_0 (opt)	
Block	Maximum	9	ISIG1	ISIG2				y_0 (opt)	
Block	Minimum	10	ISIG1	ISIG2				y_0 (opt)	
Block	Time delay	11	ISIG1		τ			y_0[a]	

continues on next page

Table 6.8.3 – continued from previous page

Type	Variable	JTYPE	J1SIG	J2SIG	F1SIG	F2SIG	F3SIG	F4SIG	F5SIG
Block	Natural logarithm	12	ISIG1		g			y_0 (opt)	
Block	Exponentiation	13	ISIG1	ISIG2	g			y_0 (opt)	
Block	Velocity limiter	14	ISIG1	-	V_{down}	V_{up}	g	y_0 (opt)	
Block	AND	15	ISIG1	ISIG2				y_0 (opt)	
Block	OR	16	ISIG1	ISIG2				y_0 (opt)	
Block	NOT	17	ISIG1					y_0 (opt)	
Block	Comparator	18	ISIG1	ISIG2				y_0 (opt)	
Block	Sample and hold	19	ISIG1	ISIG2				y_0 (opt)	
Block	JK flip-flop	20	ISIG1	ISIG2				Q_o	
Block	Constant	21			g			y_0 (opt)	
Block	Sine	22	ISIG1		g_1	g_2	g_3	y_0 (opt)	
Block	Variable Lag Compensator	23	ISIG1	ISIG2	g			y_0 (opt)	e_z
Control	Reactivity, $	-1	Signal number used					y_0 (opt)	e_z
Control	Pump motor torque, normalized	-2	Signal number used	Pump number				y_0 (opt)	e_z
Control	Steam generator, feedwater mass flowrate	-3	Signal number used	Steam generator number				y_0 (opt)	e_z
Control	Steam generator, feedwater enthalpy	-4	Signal number used	Steam generator number				y_0 (opt)	e_z
Control	Steam generator, steam mass flowrate	-5	Signal number used	Steam generator number				y_0 (opt)	e_z

continues on next page

Table 6.8.3 – continued from previous page

Type	Variable	JTYPE	J1SIG	J2SIG	F1SIG	F2SIG	F3SIG	F4SIG	F5SIG
Control	Sodium valve loss coefficient	-6	Signal number used	Valve number				y_0 (opt)	e_z
Control	Steam generator, steam pressure	-7	Signal number used	Steam generator number				y_0 (opt)	e_z
Control	Air dump heat exchanger, air mass flowrate	-8	Signal number used	Air dump heat exchanger number				y_0 (opt)	e_z
Control	*Simulation trip*	-10	Signal number used	Latching option (0, Unlatched; 1, Latched)				y_0 (opt)	e_z

[a] Not required if steady state solution finder is used, J1SIG(999)=1.

6.8.4 Data Cards

```
form = "(2I6,5F12.0)"
read(*,form) LOC, N, VAR1, VAR2, VAR3, VAR4, VAR5
```

A data card appearing in the control system block has a format identical to the standard SAS4A/SASSYS-1 data card used in floating-point input blocks and is processed in the same way. This means that *free-formatted input* may be used as an alternative to the fixed format shown above.

Data cards are used to construct demand tables, function generator tables and to supply solution convergence parameters. These quantities and their storage locations are defined in Table 6.8.4.

[1] Compressible volume number
[2] Liquid segment number
[3] Pump number
[4] Core channel number
[5] Axial location on the core channel fuel mesh.
[6] Default scale = 1.0. Returned value is the result of multiplying the measured value by *scale* and then adding *offset*.
[7] Default offset = 0.0. Returned value is the result of multiplying the measured value by *scale* and then adding *offset*.
[8] Axial location on the core channel coolant mesh.
[9] If non-zero, measured signal is evaluated over the range ICH:ICH2.
[10] Signal number used as input for the first paramater of a block signal.
[11] Signal number used as input for the second paramater of a block signal.
[12] Zero crossing parameter

Table 6.8.4: Table Card Data

Location	Fortran Symbol	Definition/Comments
1	CTLTAB (J,J1SIG)	Table of normalized demand values. A table is defined when at least two entries are provided. Index J1SIG designates the table number and J is the entry number in the table. Dimension (20,100).
2001	CTLTIM (J,J1SIG)	Times for CTLTAB table. Values must be in ascending order. Dimension (20,100).
4001	CTLFNC (J,J1SIG)	Table of function generator dependent variables. A table is defined when at least two entries are provided. Index J1SIG designates the table number and J is the entry number in the table. Dimension (20,100).
6001	CTLSIG (J,J1SIG)	Table of independent variables for CTLFNC table. Values must be in ascending order. Dimension (20,100).
8001	EPSCS	Convergence parameter for dynamic blocks over a subinterval. Must be greater than zero.
8002	EPSCPL	Maximum relative change in a control signal over a subinterval. Must be greater than zero.

6.8.5 Sample Input

Listing 6.8.1 and Listing 6.8.2 show examples of input for the control system. The first listing utilizes a weighted average of the liquid temperature in four compressible volumes to control a valve and additional reactivity. (Note that 0.0245+0.7026+0.2435+0.0294 = 1.) The second listing shows how to access core channel state variables and apply scale and offset parameters. It also demonstrates the calculation of a simple average.

Listing 6.8.1: Using the Control System to Control Flow through a Valve

```
INCONT      5
!
! Use the Control System to control overflow through Valve 1
!
!         JTYPE: Signal Type
!         |    J1SIG
!ISIG     |    |    J2SIG
!   |     |    |    |          F1SIG     F2SIG     F3SIG     F4SIG     F5SIG
!   |     |    |    |          |         |         |         |         |
    1   -57    1    0                                              100.0    !␣
↪TL(CV1)
    2   -57    2    0                                              100.0    !␣
↪TL(CV2)
    3   -57    3    0                                              100.0    !␣
↪TL(CV3)
    4   -57    4    0                                              100.0    !␣
↪TL(CV4)
    5    1     1    2    0.0245    0.7026       1.0                100.0    ! 0.
↪0245*TL(CV1)+0.7026*TL(CV2)
    6    1     3    4    0.2435    0.0294       1.0                100.0    ! 0.
↪2435*TL(CV3)+0.0294*TL(CV4)
    7    1     5    6       1.0       1.0       1.0                100.0    !␣
↪TLBAR
```

(continues on next page)

(continued from previous page)

```
    8    8    7    1       1.0                                    1.0E06    !␣
↪Table of K_orf
    9   -6    8    1                                              1.0E07    ! K_
↪orf for Valve 1
   10    8    7    2       1.0                                      0.01    !␣
↪Table of Reactivity
   11   -1   10    0                                              0.01    !␣
↪Control Reactivity

  999 ! End of signal definitions

!                      Function Generator 1: Dependent Value (orifice)
!                      |          |          |          |          |
  4001     5      1.0E10      1.0E10      1.0E09      1.0E06      1.0E03
  4006     2         0.0         0.0

!                      Function Generator 1: Independent Value (TLBAR)
!                      |          |          |          |          |
  6001     5         0.0       738.5       739.0       739.5       740.0
  6006     2       740.5     10000.0

!                      Function Generator 2: Dependent Value (reactivity)
!                      |          |          |          |          |
  4021     5         0.0         0.0        0.01        0.05        0.10
  4026     2      0.1954      0.1954

!                      Function Generator 2: Independent Value (TLBAR)
!                      |          |          |          |          |
  6021     5         0.0       738.5       739.0       739.5       740.0
  6026     2       740.5     10000.0

!                      EPSCS: Convergence parameter for dynamic blocks
!                      |          EPSCPL: Maximum relative change in a control
!                      |          |       signal over a subinterval
  8001     2         0.1         0.5

END
```

Listing 6.8.2: Average of Inlet and Outlet Temperature for Channel 1, in Degrees Fahrenheit.

```
INCONT     5
!
!        JTYPE: Signal Type
!        |    J1SIG
!ISIG    |    |    J2SIG
!   |    |    |    |         F1SIG     F2SIG     F3SIG
!   |    |    |    |         |         |         |
    1 -120    1    0       1.8   -459.67             ! Coolant Inlet␣
↪Temperature, °F
    2 -123    1    0       1.8   -459.67             ! Coolant Outlet␣
↪Temperature, °F

!   |    |    |    |         |         |         |
    3    1    1    2       1.0       1.0       0.5   ! Summer/Average, °F
```

(continues on next page)

(continued from previous page)

```
  999
!                       EPSCS: Convergence parameter for dynamic blocks
!                       |           EPSCPL: Maximum relative change in a control
!                       |           |       signal over a subinterval
  8001     2          0.1          0.1
END
```

CHAPTER

SEVEN

BALANCE OF PLANT

7.1 Introduction

A balance-of-plant thermal/hydraulic model has been developed for use with the SASSYS-1 liquid metal reactor systems analysis code. This model expands the scope of SASSYS-1 so that the code can explicitly model the waterside components of a nuclear power plant. Previously, only the water side of the steam generators could be modelled, with the remainder of the water side represented by boundary conditions on the steam generator. This chapter is organized with a structure that reflects the three major areas of the balance-of-plant thermal/hydraulics module: 1) the network model, 2) the steam generator model, and 3) the component models.

The balance-of-plant network thermal/hydraulics model represents the water side as a network of components, similar to the representation used by SASSYS-1 for the sodium side of the plant. The model will handle subcooled liquid water, superheated steam, and saturated two-phase fluid. With the exception of heated flow paths in heat exchangers, the model assumes adiabatic conditions along flow paths. This assumption simplifies the solution procedure while introducing very little error for a wide range of reactor plant problems.

The balance-of-plant steam generator thermal/hydraulics model is totally new, and completely replaces the previous SASSYS-1 steam generator model. A number of modeling improvements have been implemented in the new steam generator model to achieve better accuracy, capability, and computational performance. First, variable spatial nodalization has been introduced to permit more accurate estimates of heat fluxes. Second, an improved treatment of heat transfer regime transitions is employed to promote numerical stability. Third, donor cell spatial differencing is used with a semi-implicit time-differencing scheme to add further numerical stability and to allow larger time stops. Fourth, the steady-state initialization process has been improved to accommodate a wider variety of heat transfer and flow regimes. Finally, the new steam generator model contains a preliminary method for variable time step selection.

The steam generator model simulates the two most likely configurations for LMR plants with their high core outlet/inlet temperatures. The two configurations are: 1) the once-through system where the superheater is an integral part of the evaporator, and 2) the external recirculation type where the balance-of-plant components are used to connect the separate superheater to the evaporator through the steam drum.

The balance-of-plant components thermal/hydraulics models extend the scope of the network model, to treat nonadiabatic and two-phase conditions along flow paths, and to account for work done across the boundaries of compressible volumes. Simple conservation balances and extensive component data in the form of correlations constitute the basis of models of heaters (deaerators, steam drums, condensers, reheaters, flashed heaters, drain coolers, desuperheating heaters, and desuperheater/drain coolers), turbines, and relief valves. Except for the turbine nozzle, the mass and momentum equations for the component models are the same as the network equations. The component energy equations contain a heat source due to energy transfer across a flow boundary or to work done through a shaft. To handle two-phase conditions, the equation of state is expressed differently for each phase in terms of the quality and separate intensive properties.

The next three sections describe the thermal-hydraulic network, steam generator, and component models.

7.2 Network Thermal/Hydraulic Model

7.2.1 Discretization of the Balance of Plant

SASSYS-1 represents the balance of plant as a network of one-dimensional flow paths which are joined at flow junctions. Junctions are of two types; those in which there is no separation of liquid and vapor (in which case perfect mixing is assumed) and those in which liquid and vapor are strictly separated (in which case instantaneous separation and saturation conditions are assumed). As a result of using this representation, the mass, momentum, and energy equations that describe the system can be reduced to one-dimensional forms. The network of junctions and paths is just a discretization of the balance of plant using a non-uniform spatial mesh; the network is perhaps best described as a nodalization of the plant. The momentum equation is solved along each flow path, and the mass and energy equations are solved at each flow junction. Flow is assumed uniform throughout each flow path. Because the junctions are regions of fixed volume and are used in part to model compressibility effects in the system, they are given the name compressible volumes. Flow paths are known simply as flow segments.

The balance of plant is a collection of several types of components: pipes, valves, check valves, pumps, heaters, inlet and outlet plena, and piping junctions. The steam generator is intentionally omitted form this list, since it is treated as a separate model (Section 7.3). Components which primarily affect mass flow rate and pressure drop in a flow segment are best described through the momentum equation and are modelled as sections of flow segments; these sections are called flow elements. The cross-sectional area is constant throughout a given flow element. Element types include pipes, valves, check valves, and pumps. Flow segments then become strings of one or more flow elements; for example, a flow segment might consist of a length of pipe (element 1), followed by a pump (element 2), followed by a check valve (element 3). The flow would be the same in all three elements, but the geometry could differ from element to element.

Components which join two or more flow segments are best described through the mass and energy equations and are modelled as compressible volumes; these include inlet and outlet plena, piping junctions, and open heaters. Closed heaters must be described through a combination of flow elements and a compressible volume (Section 7.4).

More detailed information about constructing a nodalization of a plant will be presented in Section 7.2.7.

7.2.2 Analytical Equations

The general analytical equations are

$$\text{mass: } \frac{\partial \rho}{\partial t} = -\left(\boldsymbol{\nabla} \cdot \rho \mathbf{u}\right) \tag{7.2-1}$$

$$\text{momentum: } \frac{\partial}{\partial t}\left(\rho \mathbf{u}\right) = -\left[\boldsymbol{\nabla} \cdot \rho \mathbf{u}\mathbf{u}\right] - \boldsymbol{\nabla} P - \left[\boldsymbol{\nabla} \cdot \boldsymbol{\tau}\right] + \rho \mathbf{g} \tag{7.2-2}$$

$$\text{energy: } \frac{\partial}{\partial t}\left(\rho \hat{E}\right) = -\left(\boldsymbol{\nabla} \cdot \rho \overline{E} \mathbf{u}\right) - \boldsymbol{\nabla} \cdot \mathbf{q} - \left[\boldsymbol{\nabla} \cdot P \mathbf{u}\right] - \left[\boldsymbol{\nabla} \cdot \left(\boldsymbol{\tau} \cdot \mathbf{u}\right)\right] \tag{7.2-3}$$

These can be simplified by making the following assumptions:

1) One-dimensional flow,
2) Neglect the work done by viscous forces on a compressible volume (this is the $\boldsymbol{\nabla} \cdot \left(\boldsymbol{\tau} \cdot \mathbf{u}\right)$ term in Eq. (7.2-3)),
3) Neglect kinetic energy and gravitation energy,
4) The viscous term in the momentum equation can be expressed as $\frac{F \rho u |u|}{2}$.

In addition, if the total energy (which is now assumed to equal the internal energy) is expressed in terms of the enthalpy, the mass, momentum, and energy equations take the simpler forms

$$\text{mass: } \frac{\partial \rho}{\partial t} = -\frac{\partial \rho u}{\partial z} \tag{7.2-4}$$

$$\text{momentum: } \frac{\partial \rho u}{\partial t} = -\frac{\partial \rho u^2}{\partial z} - \frac{\partial P}{\partial z} - F\frac{\rho u\,|u|}{2} + \rho g cos\theta \tag{7.2-5}$$

$$\text{energy: } \frac{\partial \rho h}{\partial t} = -\frac{\partial \rho h u}{\partial z} - \frac{\partial q}{\partial z} + \frac{\partial P}{\partial t} \tag{7.2-6}$$

The system is closed by using an equation of state; for single-phase fluid, this takes the form

$$\frac{\partial v}{\partial t} = \left(\frac{\partial v}{\partial P}\right)_h \frac{dP}{dt} + \left(\frac{\partial v}{\partial h}\right)_P \frac{dh}{dt} \tag{7.2-7}$$

while for two-phase fluid, the equation of state is

$$\frac{\partial v}{\partial t} = \left(\frac{\partial v}{\partial P}\right)_h \frac{dP}{dt} + \left(\frac{\partial v}{\partial x}\right)_P \frac{dx}{dt} \tag{7.2-8}$$

The coefficient F in the momentum equation needs further explanation. The term $F\rho|u||/2$ accounts for pressure losses along the flow path due to friction, bends, area changes, and orifices or baffles. The coefficient F is the sum of three factors:

1) the frictional factor, fL/D_h, with L the flow element length, D_h the element hydraulic diameter, and f the friction factor. For turbulent single-phase flow, f is the Moody friction factor

$$f = C_1\left\{1 + \left(C_2\frac{\varepsilon}{D_h} + \frac{C_3}{Re}\right)^{C_4}\right. \tag{7.2-9}$$

where C1 - C4 are constants (see Ref. 7-1), ε is the element roughness, and Re is the Reynolds number,

$$Re = \frac{D_h|w|}{A\mu} \tag{7.2-10}$$

with μ the fluid dynamic viscosity. For laminar single-phase flow, f is calculated from

$$f = 64/Re \tag{7.2-11}$$

2) the term $f\,(L_B/D_B)\,N_B$, which accounts for pressure losses due to bends. Here, N_B is the number of bends along the flow path, and the term (L_B/D_B) is the effective length-to-diameter ratio per bend. The same value is used for (L_B/D_B) on the water side as is used on the sodium side. The friction factor f is the same one just described in 1).

3) the orifice coefficient, G_2. There is no formalism for independently calculating the initial value of G_2; in the SASSYS-1 balance-of-plant coding, this coefficient is usually calculated by the code in the steady-state initialization so as to balance the remaining terms in the momentum equation. During the transient calculation, G_2 will vary as orifices change size (e.g., a valve opening or closing). A detailed explanation of how G_2 is computed for various types of components is presented in Section 7.2.3.2 below.

The factor F is then defined as

$$F = f\,(L/D_h) + f\,(L_B/D_B)\,N_B + G_2 \tag{7.2-12}$$

In the case of two-phase flow, F is still given by Eq. (7.2-12), but now a two-phase friction correlation must be used rather than the single-phase expression of Eq. (7.2-9) and Eq. (7.2-11). The question of modelling pressure drop in a two-phase flow has been examined at length in the literature. The approach taken in the balance-of-plant

model is that set forth by Thom in Ref. 7-2, in which he describes a homogeneous equivalent model which makes use of two-phase multipliers to adjust the homogeneous gravitational, acceleration, and frictional pressure drop terms to accommodate general two-phase flow. The assumption is made in the balance-of-plant model that slip between the phases can be neglected. In this case, Ref. 7-2 shows that the gravitational and acceleration terms keep the same forms as for single-phase flow, with the appropriate mixture densities used. The same is assumed to be true for the orifice term. The frictional term is adjusted by multiplying the single-phase liquid friction factor by an empirically-determined two-phase multiplier. This multiplier adjusts the pressure drop computed for saturated liquid at a specific flow and pressure to give the pressure drop corresponding to two-phase flow at the same pressure and mass flow rate. The multiplier is a function of both quality and pressure.

Reference 7-2 presents a table of the two-phase multiplier appropriate to flow through an unheated pipe. The multiplier is given at discrete values of pressure from 250 to 3000 psi and at qualities between 0 and 1. These values were used to develop an expression to be used in the balance-of-plant model for the two-phase multiplier r_5 as a function of pressure and quality. The expression was derived using the method of least squares; it has the form

$$r_5 = 0.9124 + 0.198\overline{P} + x\left(-7.38 + 108.7\overline{P} - 4.166\overline{P}^2\right) \tag{7.2-13}$$

where the pressure P is in psi, $\overline{P} = P/250$ and x is the quality. This expression is accurate to 11% or better for all qualities and for pressures between 250 and 3000 psi; at most points, it is accurate to better than 5%. The two-phase friction factor f_{tp} is then just

$$f_{tp} = r_5 f_l \tag{7.2-14}$$

7.2.3 Discretized Equations

The analytical forms of the mass, momentum, and energy equations and the equation of state now need to be discretized over the compressible volumes and flow elements of the balance-of-plant nodalization. The result of the discretization will be a set of fully implicit equations which can be solved simultaneously for the changes in pressure, flow, and enthalpy in a timestep. All other quantities (e.g., densities, heat sources) will be computed explicitly.

The first step is to use the momentum equation to express the change over a timestep in the mass flow rate in each segment as a function of the changes in the segment endpoint pressures. Next, the mass and energy equations and the equation of state can be combined to express the change in pressure within a compressible volume as a function of the changes in the flows of all segments which are attached to the volume. If these two sets of equations are combined by eliminating the change in flow, the resulting matrix equation can be solved for the change in pressure in each compressible volume. The changes in flow and enthalpy, as well as the changes in all explicit variables, can then be determined.

7.2.3.1 Discretization of the Momentum Equation

The momentum equation is discretized segment by segment. In order to represent momentum transport correctly through a segment made up of more than one flow element, the momentum equation must be integrated along the length of the segment, giving

$$\begin{aligned}\sum_k \frac{L_k}{A_k}\frac{\partial w}{\partial t} = P_{in} - P_{out} - \sum_k \frac{w^2}{A_k^2}\left[\frac{1}{\rho_{O_k}} - \frac{1}{\rho_{I_k}}\right] \\ - \sum_k F(k)\,\frac{w\,|w|}{2\overline{\rho}_k A_k^2} - \overline{\rho} g(z_{out} - z_{in})\end{aligned} \tag{7.2-15}$$

where the summation is over all elements in a segment. The form of Eq. (7.2-15) is valid for all element types except pumps; the convective and viscous terms have a different form for pumps, and Eq. (7.2-15) is modified accordingly when a flow segment contains a pump.

The momentum equation must also be discretized over time. The right-hand side of Eq. (7.2-15) is a function only of flow and time and is the net pressure imbalance across the segment, labelled $\Delta P_{\text{net}}(w,t)$. If Eq. (7.2-15) is differenced over a timestep $\Delta t = t^{n+1} - t^n$, with $\Delta P_{net}(w,t)$ chosen at the advanced time t^{n+1} and then linearized, the result is

$$\sum_k \frac{L_k}{A_k}\Delta w = \Delta t \left\{ \Delta P_{net}\left(w,t\right) + \left[\Delta t \left(\frac{\partial \Delta P_{net}}{\partial t} \right)_w + \Delta w \left(\frac{\partial \Delta P_{net}}{\partial w} \right)_t \right] \right\} \tag{7.2-16}$$

If ΔP_{net} in Eq. (7.2-17) is replaced with the right-hand side of Eq. (7.2-15) and terms are rearranged, the result is an expression for the change in segment flow, Δw as a function of ΔP_{in} and ΔP_{out}, the changes in pressure at the segment inlet and outlet, respectively:

$$\Delta w = \frac{a_1 + \left[a_2 + \Delta t \left(\Delta P_{in} - \Delta P_{out}\right)\right]}{\left(a_0 - a_3\right)} \tag{7.2-17}$$

where

$$\begin{aligned} a_0 &= \sum_k \frac{L_k}{A_k} \\ a_1 &= \Delta t \left[P_{in}\left(t\right) - P_{out}\left(t\right)\right] + \sum_k \Delta a_1\left(k\right) \end{aligned} \tag{7.2-18}$$

$$\Delta a_1\left(k\right) = -\Delta t \left\{ \frac{w^2}{A_k^2}\left[\frac{1}{\rho_{O_k}} - \frac{1}{\rho_{I_k}}\right] - F\left(k\right)\frac{w\left|w\right|}{2\overline{\rho}_k A_k^2} - \overline{\rho}_k g \Delta z_k \right\} \tag{7.2-19}$$

$$a_2 = \sum_k \Delta t \frac{\partial}{\partial t}\left(\Delta a_1\left(k\right)\right) \tag{7.2-20}$$

$$a_3 = \sum_k \frac{\partial}{\partial w}\left(\Delta a_1\left(k\right)\right) \tag{7.2-21}$$

7.2.3.2 Further Details on Element Component Models

The discretized momentum equation, Eq. (7.2-17), models flow along a one-dimensional flow path. The path can contain bends, baffles or orifices, and the cross-sectional area of the path can vary. This flow path model serves as the basis for the models for pipes, valves, and check valves. What distinguishes these three component models is the way in which the coefficient F is varied with time. The only contributors to F which may vary with time are the friction factor and the size of a flow orifice. The friction factor is treated identically in all component models, and so the differences among component models are determined by the way in which each model calculates the orifice coefficient.

The model of a pipe assumes that the orifice coefficient calculated in the steady state is valid through the transient. Therefore, the pipe model keeps the orifice coefficient constant at all times. The valve models, on the other hand, simulate the opening and closing of the valve by varying the orifice coefficient. The pressure drop across a valve is related to the flow through the valve by the equation

$$\Delta P = \frac{w|w|}{\rho C^2 \phi^2\left(y\right)} \tag{7.2-22}$$

The functional relationship between the valve characteristic ϕ and the stem position y depends on the valve design and is input by the user through tables. The valve is opened or closed by varying the stem position; the user has the option of adjusting y directly or through the harmonic equation

$$m\frac{d^2y}{dt^2} + B\frac{dy}{dt} + ky = F_1(t) \tag{7.2-23}$$

so that y is controlled by the driver function $F_1(t)$, which is user-input. Since the pressure drop across the valve is related to the orifice coefficient G_2 by

$$\Delta P = G_2\frac{w|w|}{2\rho A^2} \tag{7.2-24}$$

where A is the flow area when the valve is fully open, the orifice coefficient can be expressed in terms of the valve calibration constant and characteristic as

$$G_2 = 2\left(\frac{A}{C\phi}\right)^2 \tag{7.2-25}$$

Thus, the valve aperture changes by recomputing the orifice coefficient each timestep, and the value of the coefficient is controlled by the stem position.

The check valve model works much the same as the standard valve model. However, there are a few differences. A check valve is normally either completely open or completely closed, whereas a standard valve can operate partially closed. A check valve changes between open or closed when a user-specified flow or pressure drop criterion within the valve is met. The valve then changes state by adjusting the orifice coefficient to a user-input value within a span of time that is also user-input. The criteria for opening and closing the valve, as well as the length of time the valve takes to open or close, must be set so as to avoid creating numerical instabilities in the calculation.

Because both valve models simulate valve closure by setting the orifice coefficient to a very large value, the flow through a valve in never actually zero. However, if the orifice coefficient is set sufficiently large, the resulting flow through the valve will be negligible.

Modelling pumps must be approached in a different way from that used in modelling pipes, valves, and check valves, since the convective and viscous terms are no longer simple functions of the mass flow rate. The head/flow relationships represented by these terms are instead described by a set of homologous pump curves. Two types of pump curves are available in the balance-of-plant model, one using polynomial fits and one using more complex functional forms. Both options are identical to the corresponding options used in SASSYS-1 for sodium pumps. See Chapter 5 for details of the pump models.

Superheaters are simply steam-filled pipes in which heat transfer is occurring along the length of the pipe. The balance-of-plant model therefore simulates a superheater as a pipe for which the orifice coefficient and the lengthwise-averaged enthalpy, temperature, and density are computed explicitly by a separate superheater model. The superheater model is similar to the steam generator evaporator model, except that no phase change occurs in a superheater. A detailed description of the superheater model is given in Section 7.3.

7.2.3.3 Discretization of the Mass and Energy Equations and the Equation of State

The derivation that follows assumes perfect mixing within a compressible volume. Volumes in which liquid and vapor are separated are discussed in Section 7.4.

The mass and energy equations and the equation of state are solved at each compressible volume. Because the fluid within each volume is assumed to be perfectly mixed, the enthalpy and pressure are uniform within a volume

(neglecting the pressure variations due to gravitational head). Therefore, the energy equation can be discretized in space by integrating Eq. (7.2-6) over each volume l and writing a separate energy equation for each volume,

$$V_l \frac{\partial (\rho_l h_l)}{\partial t} = -V_l \nabla (\rho u h)_l + Q_l + V_l \frac{\partial P_l}{\partial t} \tag{7.2-26}$$

where Q_l is the net rate at which heat enters volume l. Eq. (7.2-26) will be easier to work with if density and velocity are eliminated and the equation is rewritten in terms of flows and masses. This is accomplished by using the simple relationship between mass and density, $\rho V = m$, and recognizing that the enthalpy convection term is just the rate at which enthalpy is convected into the volume, so that

$$V_l \nabla (\rho u h)_l = -\sum_j h_j w_j \text{sgn}(j,l) \tag{7.2-27}$$

where the sum is over all segments which are attached to volume l. Substituting in Eq. (7.2-26) for the convective term and density gives the desired form of the energy equation,

$$\frac{\partial (m_l h_l)}{\partial t} = \sum_j h_j w_j \text{sgn}(j,l) + Q_l + V_l \frac{\partial P_l}{\partial t} \tag{7.2-28}$$

Now, the time derivative of the total enthalpy $m_l h_l$ needs to be expanded so that the time derivatives of the mass and the specific enthalpy can be handled separately. To do this, three operations are performed: 1) the derivative of the total enthalpy in Eq. (7.2-28) is discretized over time, 2) the advanced time terms are linearized, and 3) second-order terms are dropped. The result is

$$\frac{\partial m_l}{\partial t} h_l^n \frac{\partial h_l}{\partial t} = \sum_l h_l^{n+1} w_j^{n+1} \text{sgn}(j,l) + Q_l^n + V_l \frac{\partial P_l}{\partial t} \tag{7.2-29}$$

This form of the energy equation is a linear function of the volume mass, enthalpy, and pressure and of the enthalpies and flows from the segments attached to the volume. The heat source Q is assumed to be treated explicitly, and the enthalpy convection term is evaluated at the advanced time.

The next step is to eliminate the mass time derivative from the energy equation. This is done by using the mass conservation equation written for volume l and multiplied by the volume V_l,

$$V_l \frac{\partial \rho_l}{\partial t} = -V_l \frac{\partial}{\partial z} (\rho u)_l \tag{7.2-30}$$

The left-hand side of Eq. (7.2-30) is just the time derivative of the mass, and the right-hand side is the mass convection term, which is just the net flow into the volume, so that Eq. (7.2-30) can be rewritten as

$$\frac{\partial m_l}{\partial t} = \sum_j w_j \text{sgn}(j,l) \tag{7.2-31}$$

Eq. (7.2-31) is just the expression needed to eliminate the mass time derivative from the energy equation; when it is substituted into Eq. (7.2-29), the result is

$$m_l^n \frac{\partial (h_l)}{\partial t} = -h_l^n \sum_j w_j^{n+1} \text{sgn}(j,l) + \sum_j h_j^{n+1} w_j^{n+1} \text{sgn}(j,l) + Q_l^n + V_l \frac{\partial P_l}{\partial t} \tag{7.2-32}$$

There is one difficulty with the form of the energy equation shown in Eq. (7.2-32) – the enthalpies at the interfaces between the compressible volume and the attached flow segments are treated implicitly. For the range of problems for which this model has been developed, treating these enthalpies explicitly introduces only small errors at worst. Treating them implicitly results in a solution procedure which is unnecessarily complicated and cumbersome. Therefore, it is assumed that these enthalpies can be treated explicitly. Applying this assumption

to Eq. (7.2-32) and also finite differencing the time derivatives and linearizing the advanced time flows produces an energy equation of the form

$$m_l^n \frac{\Delta h_l}{\Delta t} = -h_l^n \sum_j \left(w_j^n + \Delta w_j\right) \text{sgn}\,(j,l) + \sum_j h_j^n \left(w_j^n + \Delta w_j\right) \text{sgn}\,(j,l) + Q_l^n + V_l \frac{\Delta P_l}{\Delta t} \tag{7.2-33}$$

Eq. (7.2-33) expresses the change in volume enthalpy as a function of the change in volume pressure and the changes in flow in the attached segments. If the change in enthalpy can be eliminated, the result will be an equation relating the change in volume pressure to the changes in the segment flows. This can then be combined with the expression derived from the momentum equation which relates the change in flow in a segment to the changes in pressure in the volumes at the segment endpoints, producing a matrix equation which can be solved for the pressure changes in the compressible volumes in the system.

The change in volume enthalpy can be eliminated from Eq. (7.2-33) by using the equation of state, Eq. (7.2-7) and Eq. (7.2-8) above. Consider first the case of a volume containing single-phase fluid. The equation of state is then Eq. (7.2-7). An expression for the change in volume enthalpy can be derived from Eq. (7.2-7) if the time derivative of the specific volume is rewritten as

$$\frac{\partial v}{\partial t} = -v^2 \left(\frac{\partial \rho}{\partial t}\right) = -\frac{v^2}{V}\frac{\partial m}{\partial t} = -\frac{v^2}{V}\sum_j \left(w_j + \Delta w_j\right) \text{sgn}\,(j,l) \tag{7.2-34}$$

Therefore, the equation of state takes the form

$$-\frac{v_l^2}{V_l}\sum_j \left(w_j^n + \Delta w_j\right) \text{sgn}\,(j,l) = \left(\frac{\partial v_l}{\partial P}\right)_h^n \frac{\Delta P_l}{\Delta t} + \left(\frac{\partial v_l}{\partial h}\right)_P^n \frac{\Delta h_l}{\Delta t} \tag{7.2-35}$$

where the remaining time derivatives have been finite differenced. Eq. (7.2-35) is another expression for the change in volume enthalpy as a function of the change in volume pressure and the changes in the segment flows and so can be substituted into the energy equation, Eq. (7.2-33), to eliminate the change in volume enthalpy, giving

$$\begin{aligned} -\frac{\left(v_l^n\right)^2}{V_l}\sum_j \left(w_j^n + \Delta w_j\right) \text{sgn}\,(j,l) &= \left(\frac{\partial v_l}{\partial P}\right)_h^n \frac{\Delta P_l}{\partial t} + \left(\frac{\partial v_l}{\partial h}\right)_P^n \frac{1}{m_l^n} \\ \times\Bigg\{ -h_l^n \sum_j \left(w_j^n + \Delta w_j\right) \text{sgn}\,(j,l) &+ \sum_j h_j^n \left(w_j^n + \Delta w\right) \text{sgn}\,(j,l) \\ &+ Q_l^n + V_l \frac{\Delta P_l}{\Delta t}\Bigg\} \end{aligned} \tag{7.2-36}$$

If the mass of volume l is expressed in terms of V_l and ρ_l and the equation is then rearranged, the result is an equation for the change in volume pressure as a function of the changes in the segment flows:

$$\begin{aligned} \Delta P_l = -\Delta t\Bigg\{ Q_l^n &+ \sum_j w_j^n \left[h_j^n - h_l^n + v_l^n \left(\frac{\partial h_l}{\partial v}\right)_P^n\right] \text{sgn}\,(j,l) \\ &+ \sum_j \Delta W_j \left[h_j^n - h_l^n + v_l^n \left(\frac{\partial h_l}{\partial v}\right)_P^n\right] \text{sgn}\,(j,l)\Bigg\} \Bigg/ \\ &\left[V_l \left(1 + \frac{1}{v_l^n}\left(\frac{\partial v_l}{\partial P}\right)_h^n \left(\frac{\partial h_l}{\partial v}\right)_P^n\right)\right] \end{aligned} \tag{7.2-37}$$

If volume l contains two-phase fluid, the equation of state must be expressed by Eq. (7.2-8), and the resulting equations are more complex. In the case of a two-phase fluid, the specific volume is a function of pressure and

quality, rather than pressure and enthalpy, and so Eq. (7.2-8) cannot be used directly to derive an expression for the change in enthalpy. Instead, the following approach must be taken.

The saturated liquid specific volume and enthalpy can be written respectively as

$$v_l = v_f + (v_g - v_f)\, x_l = v_f + v_{fg} x_l \tag{7.2-38}$$

and

$$h_l = h_f + h_{fg} x_l \tag{7.2-39}$$

Differencing Eq. (7.2-38) with respect to time gives

$$\frac{dv_l}{dt} = \frac{dv_f(l)}{dt} + x_l \frac{dv_{fg}(l)}{dt} + v_{fg}(l) \frac{dx_l}{dt} \tag{7.2-40}$$

Since the saturated liquid and vapor specific volumes are functions only of pressure, Eq. (7.2-40) can be rewritten in terms of the time derivative of pressure,

$$\frac{dv_l}{dt} = \frac{dP_l}{dt}\frac{dv_f(l)}{dP} + x_l \frac{dP_l}{dt}\frac{dv_{fg}(l)}{dP} + v_{fg}(l) \frac{dx_l}{dt} \tag{7.2-41}$$

Similarly, Eq. (7.2-39) can be expressed as

$$\frac{dh_l}{dt} = \frac{dP_l}{dt}\frac{dh_f(l)}{dP} + x_l \frac{dP_l}{dt}\frac{dh_{fg}(l)}{dP} + h_{fg}(l) \frac{dx_l}{dt} \tag{7.2-42}$$

The time derivative of quality can be eliminated between Eq. (7.2-41) and Eq. (7.2-42) to give

$$\begin{aligned} \frac{dv_l}{dt} = -\frac{v_l^2}{V_l}\sum_j w_j^{n+1}\,\mathrm{sgn}(j,l) = \frac{dP}{dt}\Bigg[\frac{dv_f(l)}{dP} + x_l\frac{dv_{fg}(l)}{dP} \\ - \left(\frac{dh_f(l)}{dP} + x_l\frac{dh_{fg}(l)}{dP}\right)\frac{v_{fg}(l)}{h_{fg}(l)}\Bigg] + \frac{dh_l}{dt}\frac{v_{fg}(l)}{h_{fg}(l)} \end{aligned} \tag{7.2-43}$$

Eq. (7.2-43) can be used in place of Eq. (7.2-8) to give an expression for the change in volume enthalpy in a two-phase volume. This expression can then be used in the same manner that Eq. (7.2-35) was used above, producing an equation for the change in volume pressure as a function of the changes in the flows in the attached segments:

$$\begin{aligned} \Delta P_l = -\Delta t\Bigg\{Q_l^n + \sum_j w_j^n\left[h_j^n - h_l^n + v_l^n\frac{h_{fg}(l)}{v_{fg}(l)}\right]\mathrm{sgn}(j,l) \\ + \sum_j \Delta w_j\left[h_j^n - h_l^n + v_l^n\frac{h_{fg}(l)}{v_{fg}(l)}\right]\mathrm{sgn}(j,l)\Bigg\}/ \\ \Bigg\{V_l\Bigg(1 + \frac{1}{v_l^n}\left[\left(\frac{dv_f(l)}{dP}\right)^n + x_l\left(\frac{dv_{fg}}{dP}\right)^n\right]\frac{h_{fg}(l)}{v_{fg}(l)} \\ - \left[\left(\frac{dh_f(l)}{dP}\right)^n + x_l\left(\frac{dh_{fg}}{dP}\right)\right]\Bigg)\Bigg\} \end{aligned} \tag{7.2-44}$$

Eq. (7.2-44) for two-phase volumes is equivalent to Eq. (7.2-37) for single-phase volumes.

If the terms

$$\begin{aligned} \mathrm{DHDN} &= \left(\frac{\partial h_l}{\partial v}\right)_P^n \text{ for } l \text{ a single-phase volume} \\ &= \frac{h_{fg}(l)}{v_{fg}(l)} \text{for } l \text{ a two-phase volume} \end{aligned} \tag{7.2-45}$$

and

$$\begin{aligned}\text{DENOM} &= V_l\left[1+\frac{1}{v_l^n}\left(\frac{\partial v_l}{\partial P}\right)_h^n\left(\frac{\partial h_l}{\partial v}\right)_P^n\right] \text{ for } l \text{ a single-phase volume}\\ &= V_l\Bigg\{1+\frac{1}{v_l^n}\left[\left(\frac{dv_f(l)}{\partial P}\right)^n + x_l\left(\frac{dv_{fg}(l)}{dP}\right)^n\right]\frac{h_{fg}(l)}{V_{fg}(l)}\\ &\quad -\left[\left(\frac{dh_f(l)}{\partial P}\right)^n + x_l\left(\frac{dh_{fg}}{dP}\right)^n\right] \text{ for } l \text{ a two-phase volume}\end{aligned} \tag{7.2-46}$$

are defined and substituted into Eq. (7.2-37) and Eq. (7.2-44), a single expression for the combined mass and energy equation can be written as

$$\begin{aligned}\Delta P_l = -\Delta t\Bigg\{&Q_l^n + \sum_j w_j^n\left[h_j^n - h_l^n + v_l^n\left(\text{DHDN}(l)\right)\right]\text{sgn}(j,l)\\ &+\sum_j \Delta w_j\left[h_j^n - h_l^n + v_l^n\left(\text{DHDN}(l)\right)\right]\text{sgn}(j,l)\Bigg\} \,/\, \text{DENOM}(l)\end{aligned} \tag{7.2-47}$$

7.2.3.4 Further Details Concerning Energy Transfer

Two aspects of energy transfer still have not been discussed in detail. One of these is the calculation of the heat source Q in a heater volume, and the other is the calculation of the enthalpy distribution along an unheated flow segment. These two topics will now be taken up.

7.2.3.4.1 A Simple Heater Model

Because of the wide variety of types of heaters which can be part of a power plant water side, the balance-of-plant model includes nine different heater representations. Eight of these are detailed models of specific types of heaters; these are the focus of most of Section 7.4, and the reader is referred to Section 7.4 for information on them. The ninth model is a very simple, basic one which is retained as an option for the user who wishes to include a heater component in a plant layout but does not want the level of detail involved in using one of the other heater models. This simple model will now be described.

The heat source Q is computed from Newton's law of cooling in the simple heater model. The fluid on the primary side of a heater is modelled as a compressible volume and is in thermal contact with a fluid on the heater secondary side. The heater is assumed to have subcooled liquid as the primary side fluid. The secondary side is assumed to be at a uniform temperature T_s, where T_s is time-dependent and is input by the user in the form of a table of temperature versus time. Neither the secondary side nor the tube separating the two sides is modelled in detail; rather, they are lumped together as a single thermal resistance which is constant throughout the transient. On the primary side, the heat transfer coefficient is computed from the Dittus-Boelter equation in the steady state and is updated during the transient from the Dittus-Boelter correlation (see Section 7.7.2).

The initialization of the heater model in the steady state begins by finding the steady-state heat source from the net rate of enthalpy convection into the heater volume,

$$Q(0) = \sum_j h_j w_j \tag{7.2-48}$$

where the summation is over all the segments which are attached to the volume. The total heat transfer coefficient for the heater can then be found from Newton's law of cooling as

$$h_{tot}(0) = \frac{Q(0)}{A_{ht}(T_s - T_p)} \tag{7.2-49}$$

Here, A_{ht} represents the heat transfer area between the primary side and the remainder of the heater and T_p is the temperature on the primary side (which is just the temperature of the compressible volume). The Dittus-Boelter equation can now be used to calculate the primary side heat transfer coefficient $h_p\,(0)$,

$$h_p\,(0) = 0.023 \frac{k_p\,(0)}{D_{hp}} \left(Re\,(0)\right)^{0.8} \left(P_f\,(0)\right)^{0.4} \tag{7.2-50}$$

The variable k_p is the thermal conductivity of the primary side fluid, and D_{hp} the hydraulic diameter of the primary side. Since now both the total heat transfer coefficient and the primary side one are known, the coefficient h_s for the remainder of the heater can be found from the relationship

$$\frac{1}{h_s} = \frac{1}{h_{tot}} - \frac{1}{h_p} \tag{7.2-51}$$

This coefficient will then stay fixed throughout the transient. This completes the initialization of the heater model.

During the transient, the heat source is treated explicitly, so the heater model coding consists of computing a new value for Q after the pressures, flows, and enthalpies have been updated. To do this, first the current primary side heat transfer coefficient is found from the Dittus-Boelter correlation, then Eq. (7.2-51) is used to update the total heat transfer coefficient. The new value of Q can then be found from Eq. (7.2-49).

7.2.3.4.2 Modelling the Enthalpy Distribution Along a Nonheated Segment

So far, all discussion of calculation of enthalpy has been limited to the enthalpy of fluid in compressible volumes or at the interface between a volume and a segment. However, the enthalpy distribution along a segment must also be computed. This section describes the model used to find the enthalpy distribution within a nonheated segment; Section 7.3 discusses the calculation of enthalpy along the heated segment in a superheater, and Section 7.3 describes how the enthalpy distribution is calculated in the heated segment sections of heaters.

The fundamental assumption of this model is that flow within each unheated segment is adiabatic, so that there is no exchange of heat with the pipe walls, pump impellers, etc. This assumption should be valid for many types of problems and allows the enthalpy distribution to be calculated from a simple, efficient algorithm in which enthalpy is transported from one compressible volume to another. The enthalpy at any point in a segment can be found just by tracking the movement of fluid through the segment over each timestep.

Two assumptions are made in order to initialize the enthalpy distribution along each segment. First, it is assumed that the temperature is constant along each subcooled liquid segment and that this temperature is equal to the temperature of the compressible volume at the segment inlet. Second, for two-phase and vapor segments, it is assumed that the enthalpy along the segment is constant and is equal to the enthalpy of the compressible volume at the segment inlet.

During the transient calculation, the algorithm first solves for segment flows and compressible volume enthalpies and pressures, then invokes the enthalpy transport model to update the segment enthalpy distributions. The enthalpy transport model begins by finding the average over the timestep of the fluid velocity in each segment from the expression

$$\overline{u} = \frac{w^{n+1/2}}{\overline{(\rho A)}} \tag{7.2-52}$$

where

$$\overline{(\rho A)} = \frac{\overline{\rho}}{L} \sum_j A_j L_j \tag{7.2-53}$$

The summation is over all the elements in the segment, with A_j the cross sectional area of an element, L_j the element length, and L the length of the segment. It is assumed that the fluid which was in the segment at the start

of the timestep has all travelled a distance $\overline{u}\Delta t$ over the timestep. The algorithm therefore shifts the enthalpy distribution by $\overline{u}\Delta t$, plus accounts for fluid which has entered the segment inlet and fluid which has left from the segment outlet. In the case of the first timestep in the transient, this procedure results in information about the enthalpy at three points: the inlet, the outlet, and at $\overline{u}_1\Delta t_1$, the point corresponding to fluid which was at the inlet at the start of the timestep. After the second timestep, the enthalpy will be known at four points along the segment: the inlet, the outlet, at $\overline{u}_2\Delta t_2$ from the inlet, and at $(\overline{u}_1\Delta t_1 + \overline{u}_2\Delta t_2)$ from the inlet, assuming that this last point has not travelled so quickly as to pass out the segment outlet. The result is that the code tracks the progression of a set of points through each segment each timestep and interpolates between the points if enthalpy values are needed which fall between points (such as at the ends of the elements).

If fluid is moving fairly quickly through a segment, the number of points being tracked will remain small. However, if the flow becomes low enough, a situation could develop in which the algorithm is tracking an unnecessarily large number of points. Therefore, for each segment, a maximum is set on the number of points which can be tracked at one time, and a new point is not added at the end of a timestep if the point added most recently has not moved sufficiently far from the segment entrance.

In the vapor segments, simple enthalpy transport does not always provide an accurate model, since the enthalpy of a highly compressible fluid is sensitive to changes in pressure along the segment. Simple enthalpy transport might, for example, produce very inaccurate results during a transient in which the vapor moved slowly relative to the changes in pressure with time. Therefore, in the vapor segments, a second algorithm is available which assumes that the enthalpy along the length of a segment is constant in space and is identical to the enthalpy of the fluid in the compressible volume at the segment entrance. The user can set a flag to choose between enthalpy transport or constant enthalpy in vapor segments.

7.2.3.5 Boundary Conditions

Often, it is desirable to perform calculations on only a portion of a power plant and to use boundary conditions to simulate the effect of the remainder of the plant. The balance-of-plant model has two types of boundary conditions available for this purpose: a flow boundary condition and a volume boundary condition. The flow boundary condition specifies flow and enthalpy as a function of time in a special flow segment attached to a compressible volume of the user's choice. The values of flow and enthalpy are either given by the user or are controlled by the plant control system. Multiple flow boundary conditions can be applied to the system. Flow boundary conditions are treated explicitly, and so they contribute to the combined mass and energy equations, Eq. (7.2-37) and Eq. (7.2-44), only in the term

$$\Delta P = -\Delta t \sum_j w_j^n h_j^n \text{sgn}\,(j,l) \tag{7.2-54}$$

which is the second term in the numerator. The summation over j must include not only all flow segments which are attached to volume l but also any flow boundary condition attached to the volume.

A volume boundary condition is modelled as a compressible volume in which pressure, enthalpy, and quality as a function of time are either user-input or controlled by the plant control system. The volume can be attached to one or more flow segments. Multiple volume boundary conditions can be designated in a network. A volume boundary condition contributes an endpoint pressure to any flow segments to which it is attached and so affects terms a_1, a_2, and a_3 (Eq. (7.2-19), Eq. (7.2-20), Eq. (7.2-21)) in the momentum equation. It also provides inlet enthalpy and quality for any attached segments in which flow is directed out from the boundary condition.

Both flow and volume boundary conditions are updated at the start of each timestep.

7.2.4 Solution Procedure

The result of the discretization procedure described in Section 7.2.3 is a set of two equations: Eq. (7.2-17), which expresses the change in segment flow rate as a function of the changes in segment endpoint pressures, and Eq. (7.2-47), which expresses the change in compressible volume pressure as a function of the changes in the flows in each of the segments attached to the volume. These two equations can be combined into a single equation which has only changes in compressible volume pressures as variables. By substituting Eq. (7.2-17) in Eq. (7.2-47), the segment flows are eliminated, giving

$$\begin{aligned}\Delta P = -\Delta t\Bigg\{ Q_l^n + \sum_j w_j^n \left[h_j^n - h_l^n + v_l^n \text{DHDN}\left(l\right)\right] \text{sgn}\left(j,l\right) \\ + \sum_j \text{sgn}\left(j,l\right) \left[h_j^n - h_l^n + v_l^n \text{DHDN}\left(l\right)\right] \left[a_1\left(J\right) + \\ \left(a_2\left(j\right) + \Delta t\left[\Delta P_{in}\left(j\right) - \Delta P_{out}\left(j\right)\right]\right)\right] / \left[a_0\left(j\right) - a_3\left(j\right)\right] \Bigg\} \text{ / DENOM}\left(l\right)\end{aligned} \tag{7.2-55}$$

If Eq. (7.2-55) is written for all L compressible volumes in a system, the result is a set of equations, each of the form

$$\sum_{J=1}^{L} c\left(I,J\right) \Delta P_J = d\left(I\right) \tag{7.2-56}$$

The forms of the coefficients $c\left(I,J\right)$ and the source terms $d\left(I\right)$ in Eq. (7.2-56) are most clearly derived by separating out the contribution of each segment in Eq. (7.2-55). Consider a segment k, in which fluid flows from volume I to volume J. If Eq. (7.2-55) is written for volume I, the contribution of segment k to Eq. (7.2-55) can be expressed as the term

$$\begin{aligned}\Delta P_l\left(k\right) = -\Delta t\Bigg\{ - h_k^n\left(I\right) w_k^n + w_k^n \left[h_I^n - v_I^n \text{ DHDN}\left(I\right)\right] \\ - \left[h_k^n\left(I\right) - h_I^n + v_I^n \text{DHDN}\left(I\right)\right] \left[a_1\left(k\right) + a_2\left(k\right)\right] \\ + \Delta t\left[\Delta P_I - \Delta P_J\right] / \left(a_0\left(k\right) - a_3\left(k\right)\right) \Bigg\} \text{ / DENOM}\left(I\right)\end{aligned} \tag{7.2-57}$$

The term $h_k^n\left(I\right)$ is the enthalpy at the interface between segment k and volume 1. Eq. (7.2-55) can then be rewritten as

$$\Delta P_1 = -\Delta t Q_I^n \text{ / DENOM}\left(I\right) + \sum_k \Delta P_I\left(k\right) \tag{7.2-58}$$

The signs in Eq. (7.2-58) are consistent with the convention that a positive flow out of a volume is a flow in the negative direction.

Comparing Eq. (7.2-56) and Eq. (7.2-58) shows the contributions from segment k to the coefficients $c\left(I,\right)$ of the pressure changes to be

$$c_k\left(I,J\right) = -\Delta t\left(h_k^n\left(I\right) - h_I^n + \nu_I^n \text{DHDN}\left(I\right)\right) \bullet \left(-\Delta t\right) / \left[\left(a_0\left(k\right) - a_3(k)\right) \bullet \text{DENOM}(I)\right] \tag{7.2-59}$$

and

$$c_k\left(J,I\right) = -c_k\left(I,J\right) \tag{7.2-60}$$

while the contribution to the source term $d\left(I\right)$ is

$$\begin{aligned}d_k\left(I\right) = -\Delta t\Bigg\{ - h_k^n\left(I\right) w_k^n + w_k^n \left(h_I^n - v_I^n \text{DHDN}\left(I\right)\right) \\ - \left(h_k^n\left(I\right) - h_I^n + v_I^n \text{DHDN}\left(I\right)\right) \left(a_1\left(k\right) + a_2\left(k\right)\right) \text{ / } \\ \left(a_0\left(k\right) - a_3\left(k\right)\right) \Bigg\} \text{ / DENOM}\left(I\right)\end{aligned} \tag{7.2-61}$$

and so Eq. (7.2-57) can be rewritten as

$$\Delta P_I(k) = -\Delta P_J c_k(I,J) - \Delta P_I c_k(I,J) + d_k(I) \tag{7.2-62}$$

Eq. (7.2-62) can be substituted into Eq. (7.2-58) with the result, after rearrangement,

$$\Delta P_I \left(1 + \sum_k c_k(I,I)\right) + \sum_k \Delta P_{J(k)} c_k(I,J) = -\Delta t Q_I^n \,/\, \text{DENOM}(I) + \sum_k d_k(I) \tag{7.2-63}$$

Comparing Eq. (7.2-56) and Eq. (7.2-63) shows that the coefficients of the pressure changes have the form

$$c(I,I) = 1 + \sum_k c_k(I,I) \tag{7.2-64}$$

and

$$c(I,J) = \sum_{k:\,J(k)=J} c_k(I,J) \tag{7.2-65}$$

(the range given for the sum in Eq. (7.2-65) is over all segments which run between volumes I and J), while the source term is given by

$$d(I) = -\Delta t Q_I^n \,/\, \text{DENOM}(I) + \sum_k d_k(I) \tag{7.2-66}$$

If Eq. (7.2-56) is written for all L compressible volumes, an LxL matrix equation is created which can be inverted and solved for all L volume pressure changes simultaneously. Once the pressure changes are known, the change in mass flow rate in each segment can be computed from Eq. (7.2-17), and the changes in volume enthalpy can be calculated from the equation of state. The explicit quantities, such as density and heat source, can then be updated.

Note that this procedure constitutes a simultaneous solution for the changes in volume pressure, volume enthalpy, and segment mass flow rate, not just for the changes in pressure. The procedure simply takes advantage of two facts: 1) each flow change is a function of endpoint pressure changes only (as expressed by the momentum equation), and therefore the flow changes can be updated one at a time once the pressure changes have been computed, and 2) the coupling between the changes in enthalpy in neighboring volumes can be eliminated by using the equation of state to express the change in enthalpy as a function of the change in pressure and the changes in the attached segment flows, and so the enthalpy changes can be updated one at a time once the changes in pressure and flow rate are known. The three steps of solving a matrix equation for the changes in pressure, then updating the flows, then updating the enthalpies comprise a simultaneous solution for all three variables.

7.2.5 Coupling Between the Balance-of-Plant Model and the Steam Generator Model

It is important to note that the balance-of-plant model does not include the water side of the steam generator. Instead, the steam generator is modelled separately and is explicitly coupled to the balance-of-plant model. By so doing, the steam generator can be represented without use of the momentum equation (see Section 7.3), thereby significantly reducing the number of implicitly coupled momentum cells in the balance of plant. This is particularly important in a systems analysis code, where a fully-implicit solution scheme can easily result in unacceptably long running times. The omission of a momentum equation in the steam generator requires the assumptions that (1) the pressure drop across the steam generator can be neglected and that (2) the coupling between the steam generator outlet and the remainder of the balance of plant is not very strong; these assumptions are valid in a wide range of operational and accident situations in nuclear power plants.

The coupling between the balance of plant and the steam generator is accomplished by having the steam generator provide a pressure at the subcooled/two-phase interface within the steam generator and a flow from the steam generator outlet into the outlet plenum. A detailed discussion of how the steam generator model calculates the interface pressure and outlet flow is given in Section 7.3. The balance-of-plant model simulates the evaporator subcooled/two-phase interface as a pseudo-compressible volume which serves as a boundary condition volume, with the volume pressure specified by the steam generator model. The coupling is completed by having the balance-of-plant model provide the subcooled region flow and the average steam generator pressure. The subcooled region is treated as one of the flow segments in the balance-of-plant network of segments and volumes, with the steam generator model providing a region-averaged enthalpy which is used to calculate the average liquid temperature and thermodynamic properties needed in the momentum equation. Thus, the balance-of-plant model can calculate the subcooled region flow at the same time that flows in the other flow segments are computed, and it can calculate the steam generator average pressure as the linear average of the pressures in the steam generator inlet and outlet plena.

The coupling between the balance-of-plant and steam generator models requires some time averaging to stabilize the rate of change of the steam generator pressure, as well as limits on the rate of change of the subcooled zone flow; neither constraint affects the accuracy of the overall calculation. A detailed explanation of these constraints is provided in Section 7.3.

7.2.6 Implementation of the Balance-of-Plant Model in SASSYS-1

The coding which implements the mathematical model described in the preceding sections can be divided into two main parts: a steady-state initializer and a transient solution algorithm. The calculational procedure followed in each part will be described now, and then the operation of the coding subroutine by subroutine will be outlined.

7.2.6.1 Steady-State Initialization

The job of the steady-state initializer is to derive the system parameters from the data which the user has entered. The parameters to be calculated include the following:

For volumes:	the density and enthalpy are computed from the equation of state
For segments:	the length is computed from the element lengths, the endpoint enthalpies are derived from the equation of state, and the enthalpy transport arrays are set to steady-state values.
For elements:	the endpoint enthalpies are found from the equation of state, the orifice coefficient is calculated from the momentum equation, and the gravity head is computed from the average density (found from the equation of state) and the endpoint elevations
For steam generators:	the inlet enthalpy is set from the inlet plenum enthalpy, and the steam generator pressure is computed as the average of the pressures in the inlet and outlet plena
For heaters:	the heat transfer coefficients are initialized for the simple heater model; Section 7.4 discusses the initialization procedure for all other types of heaters. In all cases, the heat source term steady state value is computed
For pumps:	the pump head is computed from the pump element endpoint pressures, and the corresponding pump speed is then determined through an iterative procedure identical to that used for the sodium-side pumps (see Chapter 5)

7.2.6.2 The Transient Solution Algorithm

This algorithm solves the difference equations described earlier at each timestep. The timestep size used is based on the size of the timestep used in the sodium-side calculations by the PRIMAR subroutine. A maximum value for the balance-of-plant timestep is defined as a user-specified fraction of the PRIMAR timestep, with the balance-of-plant timestep always less than or equal to the PRIMAR timestep. However, the balance-of-plant timestep is also limited by the rates of change of the segment flows and the compressible volume pressures, as well as being limited by rates of change of the steam generator water and sodium flows, metal and sodium temperatures, water enthalpies, waterside void fractions, and waterside heat transfer coefficients. Therefore, depending upon transient conditions, the balance of plant may operate on a significantly smaller timestep than does the sodium loop. The balance of plant and the steam generator models always operate on the same timestep.

The calculation proceeds as follows:

1) For each steam generator, the feedwater enthalpy and flow and average pressure calculated by the balance-of-plant model the previous step are passed to the steam generator algorithm. If a superheater is associated with a steam generator, the superheater model algorithm is called first, then the steam generator model algorithm is used.

2) The steam generator outlet flow and enthalpy and the pressure difference between the subcooled/two-phase interface and the evaporator outlet are transferred to the balance-of-plant coding for each steam generator. For all superheaters, the superheater outlet enthalpy, average temperature, and average density are passed to the balance-of-plant model.

3) Now, the coefficients $c\,(I, J)$ and source terms $d\,(I)$ of Eq. (7.2-56) must be calculated. The first step is to compute the terms which depend only on conditions within a compressible volume and do not involve data from the segments. Primarily, this means computing the terms DHDN and DENOM in Eq. (7.2-45) and Eq. (7.2-46).

4) Next, the momentum equation coefficients a_0 through a_3 must be found for each segment (see Eq. (7.2-18) through Eq. (7.2-21)).

5) The terms $c\,(I, J)$ and $d\,(I)$ can now be assembled by traversing the nodalization segment by segment and using Eq. (7.2-64) through Eq. (7.2-66). The contributions from any flow boundary conditions, given by Eq. (7.2-54), are also included at this time.

6) The matrix equation resulting from step 5 is now solved for the compressible volume pressure changes by using Gauss-Jordan elimination.

7) Now that the changes in the volume pressures over the timestep are known, the updated pressures can be computed, and the updated segment flows can be found from Eq. (7.2-17).

8) The heat source terms for any heaters other than those represented by the simple model are updated now.

9) The new segment flows are used to update the compressible volume densities from Eq. (7.2-31). The changes in the compressible volume enthalpies can be computed from the changes in volume pressures and segment flows using Eq. (7.2-35) for single-phase volumes and Eq. (7.2-43) for two-phase volumes.

10) The enthalpy transport model is now used to calculate the current enthalpy distribution along each segment, except in segments where the user has opted for a uniform enthalpy distribution instead of enthalpy transport.

11) The new pump heads and speeds and the new gravity heads are now calculated. The heater source terms of any heaters represented by the simple heater model are also updated.

7.2.6.3 Code Operation

The calculational procedures just described are implemented by the set of subroutines and functions listed in Table 7.2.1, which includes both a brief description of the usage of each subroutine and identifies the calling subroutine except in the case of material property functions which are called from a number of different subroutines. The structure of the coding made up of these subroutines and functions is diagrammed in Figure 7.2.1 and Figure 7.2.2. The operation of this coding will now be discussed in some detail.

The progression of the calculation through the steady state initialization is diagrammed in Figure 7.2.1. The calculational sequence moves from left to right and from top to bottom in the diagram. Subroutine SSPRM4 is an initialization subroutine used by the sodium-side PRIMAR-4 module which has been modified to call the initialization subroutines for the balance-of-plant module also. SSPRM4 begins by calling RENUM, which reads the balance-of-plant input information and rearranges the nodalization to the form that the code will actually use (see Section 7.2.7.1). For volumes for which pressures and temperatures are centered as input data, RENUM also computes the volume enthalpies, and for this task it needs function FINDH. This function uses a Newton-Raphson scheme to solve an equation-of-state for enthalpy as a function of temperature and pressure. FINDH is also used by a number of subroutines in the transient calculation. When RENUM stores information concerning the evaporator model, it uses subroutines LoadSG and StoreSG to find the correct storage locations for each evaporator; similarly, when RENUM handles superheater information, it uses LoadSH and StoreSH to store the data correctly for each superheater. The relief valve initializing routine SSRVW is also called from RENUM, as are subroutines SELSRT, which orders members of each component category from smallest to largest user number (this is done in case the user numbering is not consecutive), and PRNTST, which sets flags to invoke prints of just the parameters selected by the user for printing.

Table 7.2.1: Descriptions of the Balance-of-Plant Subroutines

Routine	Description
ARF	This subroutine is used to solve the transcendental equation needed to find heater water level in heaters which are right-circular cylinders oriented so the cylinder axis is parallel to the ground. It is called from WTRDRV.
BOP_UpdateSG	This subroutine updates beginning of timestep variables in the evaporator model. It is called from INIT in the steady-state and from SGUNIT in the transient.
BOP_UpdateSH	This subroutine updates beginning of timestep variables in the superheater model. It is called from INITS in the steady-state and from SHUNIT in the transient.
CHVLW	This subroutine implements the check valve model. It tests to see whether a check valve has met the criteria for closing (if the valve is open) or opening (if the valve is closed) and sets the value of the orifice coefficient G2PW correctly before passing control to PIFLSG to compute the contribution of the valve to the segment moment equation. It is called from WTRDRV.
DTLM	This function performs the log mean temperature difference calculation when the single-node version of the steam generator model is used. It is called from INIT and SGUNIT.
FINDH	Function FINDH solves the equation of state for enthalpy as a function of temperature and pressure by using a Newton-Raphson scheme. This procedure is necessary because no correlation for enthalpy as a function of temperature and pressure is provided in the set of material properties expressions used in SASSYS.
HFlFUN	This function computes saturated liquid water enthalpy.
HGlFUN	This function computes saturated steam enthalpy.
INIT	This subroutine initializes the evaporator of the steam generator model. It is called from SSSTGN.
INITS	The superheater is initialized in this subroutine. It is called from subroutine SSSTGN.

continues on next page

Table 7.2.1 – continued from previous page

Routine	Description
LoadSG	The variables used in the steam generator model are not subscripted to accommodate more than one steam generator. Therefore, the data for each steam generator are stored in a single array and are transferred in and out as the calculation proceeds from steam generator to steam generator. LoadSG loads data into the associated steam generator common blocks; StoreSG stores common block data into the steam generator data structures. It is called from RENUM and from INIT in the steady state and from TSBOP in the transient. The subroutine which performs this transfer for the superheater variables is LoadSH.
LoadSH	This subroutine operates just as LoadSG does, but it transfers variables for the superheater model rather than the evaporator model. It is called from RENUM and INITS in the steady state and from TSBOP in the transient.
NAHT	The sodium heat transfer coefficient for a node within the evaporator is computed in this routine. It is called from INIT in the steady state and from SGUNIT in the transient.
NAHTS	The sodium heat transfer coefficient for a node within the superheater is computed in this routine. It is called from INITS in the steady state and from SHUNIT in the transient.
PIFLSG	This is the subroutine which computes the contribution made to the momentum equation by a pipe element. It is called from WTRDRV.
PLTBOP	This subroutine saves balance-of-plant data each main timestep for plotting. It is called in the steady state from SSBOP and in the transient from TSBOP.
PMPFLW	PMPFLW calculates the contribution made by a pump element to the momentum equation of the segment in which the pump is located. It is called by WTRDRV.
PMPFNW	This subroutine updates the speeds and heads of all pumps on the balance-of-plant side once the updated segment flows have been determined. It is called from WTRDRV.
PRNTST	This subroutine sets print flags so that only the parameters selected by the user for printing will be printed by subroutine WTRPRT. It is called from RENUM.
PRSH2O	PRSH2O is called only in the steady state and calculates the orifice coefficient and initial pressure drop for each element except for pumps, for which the orifice coefficient is defined to be zero. It is called from SSBOP.
RENUM	This subroutine reads the fixed-point balance-of-plant data, generates the nodalization which the code will use internally, and reads the floating-point data. It is called in the steady state from SSPRM4.
REVLW	This subroutine computes the fractional valve opening area as a function of pressure drop for a relief valve. A simple hysteresis curve is used to represent the opening area, so that the valve does not chatter when the pressure drop is near the point at which the valve opens or closes. The relief valve is modeled to open and close with a response time, which is the maximum valve opening time, to avoid numerical instabilities caused by the step changes in flow. It is called from TSRVW.
SELSRT	This subroutine orders the members of each component category by user number, from smallest to largest; this allows the code to handle cases in which the user has not used consecutive numbers for members of one or more component categories. It is called from RENUM.
SFFUNW	This function computes saturated liquid water entropy.
SGFUNW	This function computes saturated steam entropy.
SGMOM	This is the subroutine which calculates the momentum equation terms for the subcooled region in the evaporator and for the superheater. It is called from WTRDRV.

continues on next page

Table 7.2.1 – continued from previous page

Routine	Description
SGUNIT	SGUNIT is the subroutine which computes temperatures and flows in the steam generator. It is also the subroutine which feeds information from the steam generator model to the balance-of-plant model. It is called from TSBOP.
SHUNIT	Temperatures along the superheater are computed by this routine. It is called from WTRDRV.
SSBOP	This subroutine does the steady-state initialization for the balance-of-plant model. It is called from SSPRM4.
SSCFUN	This function calculates the subcooled water entropy as a function of pressure and enthalpy.
SSHFUN	This function calculates the superheated steam entropy as a function of pressure and enthalpy.
SSHTRW	Most of the initialization of any heaters is done in SSHTRW. SSHTRW computes the shell-side temperatures and/or the tube-side temperature distribution (along with the surface heat transfer area calibration factor). The heater heat source terms (and the heat source terms in the drain and/or the desuperheating region, if such regions are present) are calculated also in this subroutine. In addition, SSHTRW checks mass and energy conservation and finds the pseudo-heat conduction coefficient. It is called from SSBOP.
SSNZZL	This subroutine performs the steady-state initialization for the nozzle model. SSNZZL computes the isentropic enthalpy and the fluid density following the isentropic expansion. The steady-state nozzle velocity is calculated, and the nozzle flow area is calibrated in this subroutine. It is called from SSBOP.
SSPMPW	The steady-state initialization of the pumps in the balance of plant is done in SSPMPW. It is called from SSBOP.
SSPRM4	SSPRM4 is a PRIMAR-4 subroutine which is also used to call the steady state subroutines RENUM and SSBOP which initialize the balance-of-plant model. It is called from PRIMAR-4 driver subroutine SSTHRM.
SSRVW	This subroutine calibrates the relief valve flow area for a fully open valve according to the relief valve capacity. This is done in the steady-state balance-of-plant, when the flow through the relief valve is normally zero. It is called from RENUM.
SSTRBN	The steady-state initialization of the turbine stage is done in this subroutine. SSTRBN checks the conservation of mass and energy in the turbine stage. It also computes the turbine torque and sets the generator torque. It is called from SSBOP.
StoreSG	The variables used in the steam generator model are not subscripted to accommodate more than one steam generator. Therefore, the data for each steam generator are stored in a single array and are transferred in and out as the calculation proceeds from steam generator to steam generator. StoreSG stores common block data in the steam generator data structures, whereas LoadSG is utilized to load data into common. It is called from RENUM and from INIT in the steady-state and from TSBOP in the transient. The subroutine which performs this transfer for the superheater variables is StoreSH.
StoreSH	This subroutine operates just as StoreSG does, but it transfers variables for the superheater model rather than the evaporator model. It is called from RENUM and INITS in the steady-state and from TSBOP in the transient.
TRNSPT	TRNSPT tracks the transport of enthalpy through each segment in the balance of plant. It is called from subroutine WTRDRV.
TSBOP	This is the driver subroutine which calls the steam generator model and the balance-of-plant model. It also saves waterside plot data and calls WTRPRT to print waterside output. It is called from TSSTGN.

continues on next page

Table 7.2.1 – continued from previous page

Routine	Description
TSHTRW	The transient calculation of heater temperatures and heat source are done in TSHTRW. This subroutine updates the heater shell-side temperature and/or the tube-side temperature distribution. The heater heat source terms (and the heat source terms in the drain and/or desuperheating regions, if such regions are present) are also updated in TSHTRW. It is called from WTRDRV.
TSNZZL	The transient calculation of the nozzle velocity, the isentropic fluid enthalpy, the fluid density following the isentropic expansion, and the flow rate is done in this subroutine. It also generates the coefficients of the nozzle momentum equation for the matrix elements. It is called from WTRDRV.
TSRVW	This subroutine calls REVLW to check if the relief valve is open or not. It the relief valve is open, TSRVW generates the coefficients of the relief valve momentum equation for the matrix elements and later is called again from WTRDRV to calculate the flow rate; otherwise TSRVW is bypassed and returned to the calling subroutine WTRDRV.
TSTRBN	This subroutine updates the turbine stage parameters in the transient. TSTRBN calculates the turbine stage work term, updates the turbine torque, and adjusts the rotor angular velocity. It is called from WTRDRV.
VALVEW	The orifice coefficient used for a standard valve is computed in this subroutine. It is called from TSBOP.
WTRDRV	WTRDRV is the transient driver for the balance-of-plant calculation. It is in WTRDRV that the updated balance-of-plant parameters are computed. It is called from TSBOP.
WTRPRT	This subroutine prints the current values for the balance-of-plant parameters. The frequency with which it is called is at the user's discretion. It is called from TSBOP.

Once RENUM completes its work, SSPRM4 moves on to call SSBOP to do the remainder of the steady-state balance-of-plant initialization. This is the subroutine in which the steady-state enthalpies, gravity heads, segment lengths, etc. are computed. All the calculations necessary to complete the steady-state initialization of the balance of plant are performed in SSBOP except the initialization of the nozzles, which is done in SSNZZL; the initialization of the turbine, which is done in SSTRBN; the initialization of the heaters, which is done in SSHTRW; the calculation of the element orifice coefficients, which is done in PRSH2O, and the calculation of the pump speed and torque, which is performed in SSPMPW. Finally, SSBOP calls PLTBOP to save steady-state data for plotting.

After the balance of plant has been initialized, SSPRM4 calls SSSCLP, a sodium-side subroutine which in turn calls another sodium-side routine, SSLQSG, which then calls SSSTGN, the steam generator initialization driver. Both the superheater and evaporator models are initialized from this subroutine with calls to INITS and INIT, respectively. Section 7.3 discusses these routines in detail.

With the steady-state initialization completed, the transient calculation can begin. Figure 7.2.2 outlines the progression of the calculation through the coding. The PRIMAR-4 subroutine STEPTM calls subroutine TSBOP, which is the driver for the waterside transient calculation. After updating several parameters to the current sodium-side timestep, TSBOP calls VALVEW to update the condition of all standard valves. The code then begins to operate on the balance-of-plant/steam generator timestep. First, subroutine TSSTGN, the steam generator transient driver, is called. If there are any superheaters in the problem, TSSTGN calls SHUNIT to perform the heat transfer calculation within each superheater. Similarly, if there are any evaporators or once-through steam generators, TSSTGN calls SGUNIT for each one, then updates the parameters which constitute the evaporator/balance-of-plant interface. Both SHUNIT and SGUNIT are discussed in detail in Section 7.3.

The next step for TSBOP is to call WTRDRV, the balance-of-plant driver, to compute the balance-of-plant parameters. WTRDRV has the job of assembling the matrix equation for the compressible volume pressure

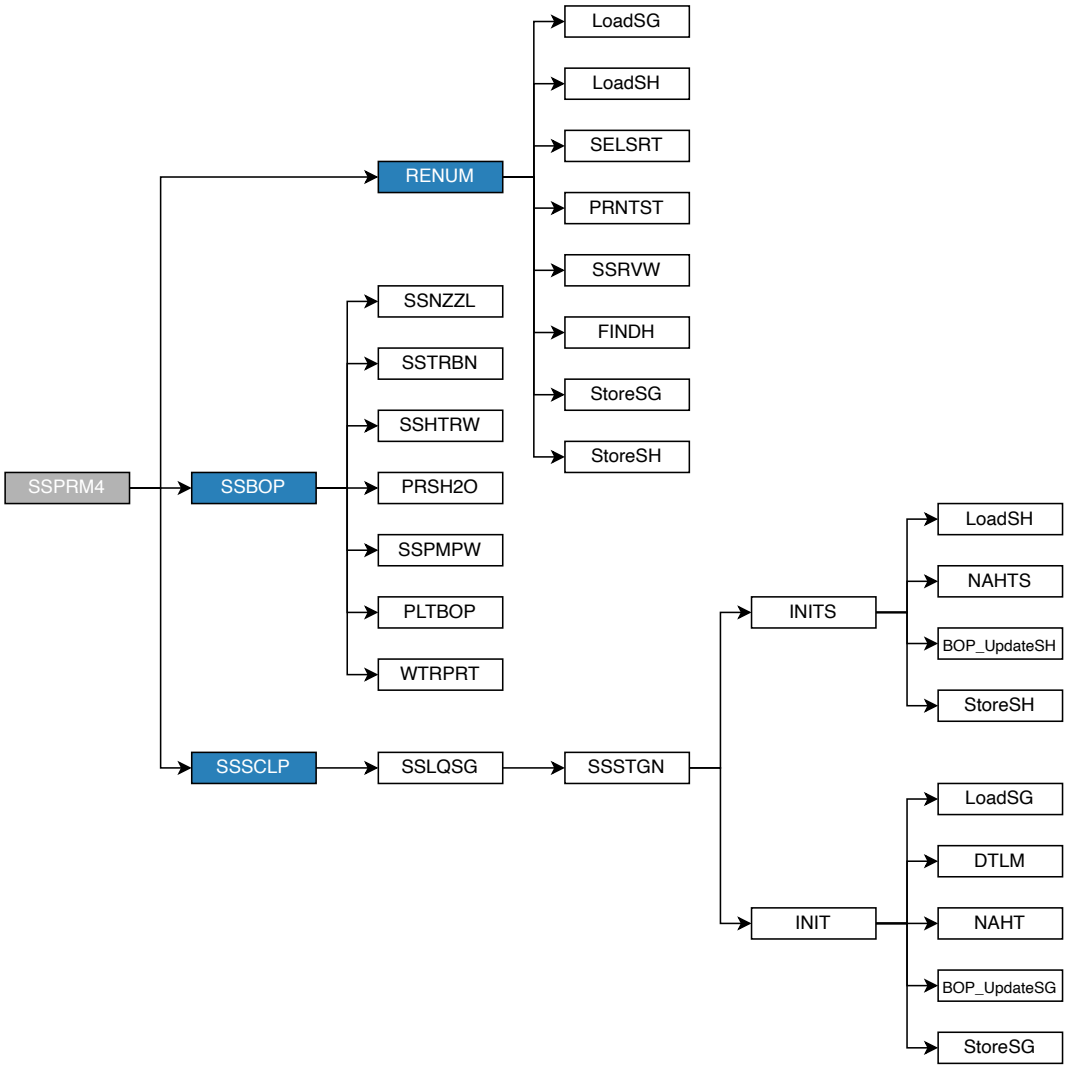


Figure 7.2.1: Balance of Plant Steady-State Program Structure

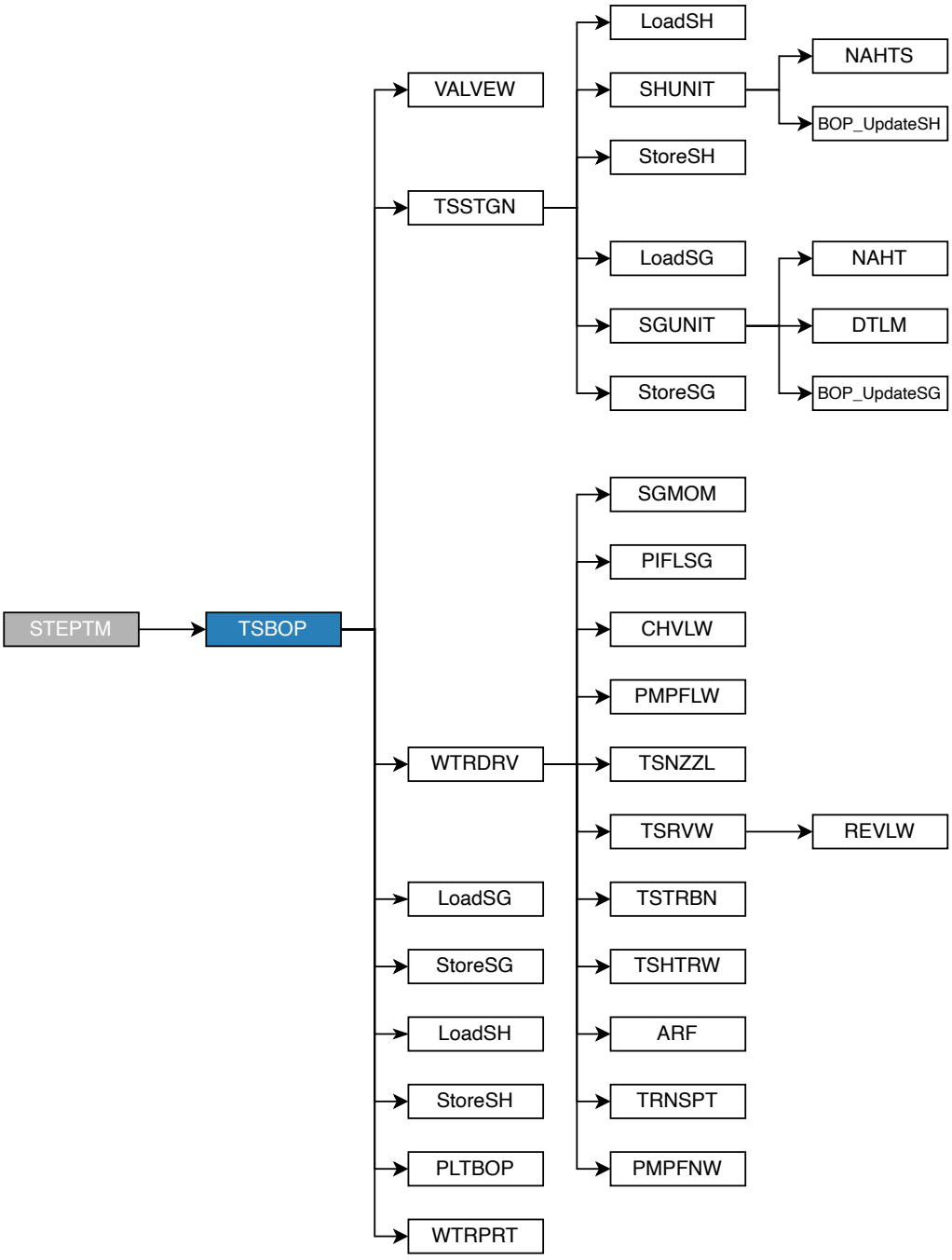


Figure 7.2.2: Balance of Plant Transient Program Structure (TSBOP)

changes, solving for the pressure changes, and then solving for the remaining balance-of-plant parameters. While most of this work is performed in WTRDRV itself, there are a number of subroutines which contribute to the calculation, as can be seen in Figure 7.2.2. The momentum equation coefficients for incompressible flow elements are computed by SGMOM for superheaters and for the subcooled region of each evaporator; by PIFLSG for pipes; by CHVLW for check valves; and by PMPFLW for pumps. For compressible flow elements, the momentum equation contributions are computed by TSNZZL for nozzles and by TSRVW for relief valves. Most calculations related to heaters other than those simulated by the simple heater model are performed by TSHTRW; simple heater model calculations take place in WTRDRV. Enthalpy transport is done by a call to TKNSPT, and PMPFNW updates pump heads and speeds. Subroutine ARF is called in conjunction with finding the new water level in some types of heaters.

Once the balance-of-plant calculations for a timestep are completed, TSBOP checks to see if the end of a sodium-side PRIMAR timestep has been reached. If so, TSBOP calls PLTBOP to save plot data and calls WTRPRT to provide a printout of the waterside parameters. Both PLTBOP and WTRPRT can be called less frequently than once each PRIMAR timestep by setting the frequency of each call through input. If the end of a PRIMAR timestep has not been reached, TSBOP increments the balance-of-plant/steam generator timestep and begins another series of calls to TSSTGN and WTRDRV.

7.2.6.4 Data Input

The input data variable names are defined in Section 7.7.1, which also lists many of the balance-of-plant variables that do not represent input quantities. A complete line-by-line description of the input data is presented in Appendix 2.2.

7.2.7 Creating a Plant Nodalization

The first step in analyzing a particular power plant using the models discussed above is to discretize the plant layout into a network composed of the component models available in SASSYS. The elements of the network (volumes, segments, etc.) must be assigned numbers for use by the code, and there are a few simple rules governing how the numbering is done, as discussed below in Section 7.2.7.1. There are also several features of the code which can help simplify the input data required and reduce the size of the problem; these are described in Section 7.2.7.2 through *7.2.7.4*.

7.2.7.1 Rules About Numbering Plant Components

Designing a nodalization for a plant layout involves going through the layout and deciding how best to simulate the plant using the component models available in the code. The plant is thereby reduced to a network of compressible volumes and flow segments, with the segments further divided into flow elements.

Once this network has been created, the nodalization is completed by numbering the volumes, segments and elements. The process of numbering is easy, as there are very few rules which must be followed in order to assemble a nodalization which SASSYS will accept. The balance-of-plant coding was designed to minimize numbering restrictions, primarily to make it easier for the user to add on to an existing nodalization. The current coding requires only that the user assign numbers in the following ranges:

For compressible volumes -- between 1 and 100

For segments -- between 1 and 100

For elements -- between 1 and 200

For pumps -- between 1 and 10

For volume boundary condition tables -- between 1 and 10

For flow boundary condition tables -- between 1 and 10

For standard valves -- between 1 and 40

For check valves -- between 1 and 40

For supersegments -- between 1 and 10

For heaters -- between 1 and 20

For legs of the balance of plant -- between 1 and 10

Steam generators and superheaters must be numbered consecutively beginning with 1 in order to be consistent with the sodium side coding. All members of a category (e.g., all segments) must be assigned unique numbers, but the same number may be used in more than one category (so there can be a segment 6 and an element 6, for example). The numbers used do not have to be consecutive, which makes it easy to add elements within a segment or add new volumes and segments to the entire system. An existing nodalization can be expanded with no renumbering of any item in the original nodalization. Members of a particular category may be assigned any numbers convenient for the user; there is no need to assign the number 1 to some category member.

This very flexible way of handling the nodalization is made possible by a section of the code which performs a renodalization internally once all the data concerning the nodalization set up by the user have been read. The user never sees the internal nodalization; when data are printed out or saved for plotting, they are numbered according to the user's nodalization. However, all the calculations performed by the code are done with the internal nodalization. The code simply renumbers the members of each category according to the order in which information was entered in the data deck (for example, if the user enters data for segment 8 first, then volume 3, then volume 15, then segment 4, the code will assign segment 8 the number 1, segment 4 the number 2, volume 3 the number 1, and volume 15 the number 2). The only exception to this is the elements; the code goes segment by segment (following the internal nodalization segment numbers) and renumbers the elements consecutively within each segment. Ordering the elements consecutively within a segment allows the program to operate more efficiently.

The following discussion will give a description of how subroutine RENUM manipulates the fixed-point input data to create a new nodalization in which each type of component is consecutively numbered starting at number one. There is also extensive documentation of this process within RENUM in the form of comment cards, and anyone with a need to understand this section of the coding in detail should refer to these comment cards.

Once all the fixed-point data are read in, and before any floating-point data are entered, RENUM performs the renodalization. Part of the renodalization process is the creation of arrays which translate between the user's numbering scheme and the scheme resulting from the renodalization. The floating-point data are not entered until after these translator arrays are created so that all floating-point arrays can be automatically ordered according to the renodalization numbering scheme as the data are read from the input file.

The goal of the renodalization is to renumber all components so that the calculations performed in solving the mass, momentum, and energy equations can be executed as efficiently as possible. If the problem is not divided into two or more legs, components of each type are simply numbered in the order in which they were entered in the fixed-point data block (so that, for example, the first element entered becomes element 1, the first volume entered becomes volume 1, etc.). If the problem is divided into two or more legs, components of each type are grouped by leg, with all components of a given type numbered consecutively within a leg in the order in which they were read in the fixed-point data block. The user must specify the order in which legs are to be arranged through the variable LEGORD.

As the fixed-point data are entered, RENUM keeps count of the total number entered of each component type. The first renodalization task it performs is to add the subcooled region of each steam generator to the total number of flow segments which were read in through input. The code then turns to the compressible volume fixed-point data. These are stored in a temporary array at the time the data are read. RENUM now marches through volume

by volume, separating volume boundary conditions from the rest and initializing boundary condition variables both for volumes which are volume boundary conditions and volumes which are attached to flow boundary condition segments. The fixed-point volume data are stored in the correct permanent arrays in the order in which the volumes were read in.

The legs of the loop are now renumbered beginning at one in the order specified by the user in array LEGORD. The array LEGBCK, which translates the user's number for a particular leg into the number assigned by the code, is generated; LEGORD is the translator array from the code numbering to the user's numbering. Once the legs are reordered, the compressible volumes (excluding volume boundary conditions) can be renumbered so that the volumes in leg 1 are numbered consecutively beginning with the number one, in the order in which they were entered in the fixed-point data input; the volumes in leg 2 are ordered consecutively following those in leg 1, etc. The arrays NCVIN and NCVOUT, which mark the first and last volumes in each leg, are set at this time, and the arrays NCVBCK, which translates from the user's numbering of volumes to the code's numbering, and NCVTRN, which translates from the code's numbering to the user's numbering, are both rearranged to be consistent with the renumbering of the volumes by leg. The remaining fixed-point volume input arrays, NTPCVW, NCVBCW, NLGCVW, and NQFLG are also rearranged.

At this point, the code turns to the flow boundary conditions and to the outlet flows passed by the evaporator model to the balance-of-plant coding. The flow boundary condition segments are numbered consecutively leg by leg, and the array JCVW, which specifies the numbers of the compressible volumes at the segment endpoints, is set for each boundary condition segment. The arrays ISGIN and ISGOUT, which designate the first and last boundary condition segments in each leg, are also set at this time. This entire process is then repeated for the segments representing each evaporator outlet flow.

The boundary condition volumes and the evaporator subcooled/two-phase interfaces are next. The code treats the interfaces as additional boundary condition volumes, with the evaporator model providing the thermodynamic parameters at each interface. RENUM first numbers the boundary condition volumes consecutively, with the numbers beginning immediately after the remaining volumes, then numbers the subcooled/two-phase interface in each evaporator in the order of numbering of the evaporators. The total number of volumes in the problem, then, is the sum of the number of interior volumes, plus the number (if any) of boundary condition volumes, plus the number of evaporator subcooled/two-phase interfaces.

After testing to see that the total numbers of volumes, segments, elements, and pumps do not exceed the dimensions of the arrays associated with these components, the code reorders several fixed-point arrays of volume-related parameters, including JCVW, NLGCVW, NBCCVF, NBCCVP, and ICVSGN. JCVW is also set for the evaporator outlet flow segments at this point. RENUM then moves on to renumbering the segments, grouping them by leg as was done for the volumes. ISGIN and ISGOUT are set here, too, to flag the first and last segment in each leg. Once the segments are renumbered, the code revises all fixed-point segment-related arrays (such as NODMAX, the maximum number of enthalpy transport nodes allowed in a segment) so that these arrays reflect the revised segment numbering rather than the user's original numbering. The elements can now be renumbered, and this is done segment by segment, with elements numbered consecutively within a segment.

The final step in creating the revised nodalization is to generate for each volume the arrays NSEGCV, ISEGCV, and ISGNCV, which give, respectively, the number of segments attached to the volume, the numbers of those segments, and the direction of flow in each attached segment. RENUM then proceeds to read in the balance-of-plant floating-point data.

7.2.7.2 Supersegments

Frequently, the best way to initialize regions containing superheated vapor is to assume that the enthalpy is constant throughout the region. However, the input data available are usually in terms of pressures and temperatures, not enthalpy, and so setting the enthalpy constant in all the volumes and segments making up a region can require some extra calculation on the part of the user and can also result in some input data inconsistencies if the user does not have available the same equation-of-state functions as are used by the code. This difficulty is easily resolved by making use of an input device called a supersegment. A supersegment is a chain of compressible volumes and flow segments, beginning and ending with a volume. The code sets the steady-state enthalpy throughout the chain, up to but not including the terminating volume, to the value of the enthalpy in the volume at the entrance to the supersegment. Up to ten supersegments can be assigned to a nodalization, and a volume can be the terminus of more than one supersegment. Supersegments provide an option which can simplify input data preparation for the user.

7.2.7.3 Legs of the Nodalization

A leg of the nodalization is the set of all volumes and segments contained between two boundary conditions on the water side. A boundary condition in this case can be a flow or volume boundary condition, or it can be an interface between a balance-of-plant component and the steam generator. Any nodalization of a plant can be considered a single leg, but in some cases, it is possible to break the plant up into more than one leg. This has the advantage of reducing the dimension of the matrix equations which must be solved for the changes in volume pressures (i.e., two or more smaller matrix equations are solved rather than one large one). Such a reduction can be very important if the code is run on a scalar machine; because the code vectorizes well, reducing the size of the matrix is of much less importance if a vector machine is used.

As an example of how to divide a problem into more than one leg, consider a case in which the balance of plant is modelled as beginning at a volume boundary condition, proceeding through a series of pumps, heaters, pipes, and valves up to a steam generator, then finishing past the steam generator with more piping ending in another volume boundary condition. This problem can be split into two legs: one from the inlet volume boundary condition to the subcooled/two-phase interface inside the steam generator, and one from the steam generator outlet plenum to the outlet volume boundary condition. Modelling the problem as two legs instead of one involves slightly more fixed-point input data but may result in significant savings in running time. A decision to try to divide a problem into multiple legs should be based on whether or not the added complexity in the nodalization is justified by any savings in running time; this will depend in part upon the machine being used. For some large problems being run on scalar machines, breaking the problem into legs may be the only cost-effective way to run a transient.

7.2.7.4 Multiplicity

Sometimes, the size of a plant nodalization can be reduced by taking advantage of symmetries which exist at least for some types of transients. The SASSYS code is equipped to take advantage of symmetries by use of a parameter called multiplicity. The operation of multiplicity in the code is best explained by a simple example.

Suppose a plant has two pumps operating in parallel, as in Figure 7.2.3. Now, some types of problems, the two pumps may behave very differently (e.g., one pump may trip and coast down while the other continues to operate), and so the momentum equation must be solved separately within each segment. However, in other problems, the pumps may operate identically, and so it would be a duplication of effort to solve the momentum equation in each segment. This is where multiplicity comes in. The two pump segments can be replaced by one segment so long as a proper accounting of the flows at the pump header and manifold is done. In this case, the correct flow contribution to each compressible volume is made by defining a multiplicity factor of two and multiplying the segment flow by this factor when solving the conservation equations. A multiplicity factor is needed at each end of any segment, since, in general, duplicate branches of a system can contain more than one

segment, and so a factor of two might be needed at one end of a segment while a factor of one might be needed at the other end.

The real value of using multiplicity comes not in a simple case, such as the pump configuration of this example, but in cases involving major branches of the balance of plant. Computing time can be reduced significantly, for instance, in the case of a plant which has two steam generators and therefore two identical branches from the feedwater pump manifold to the high-pressure turbine. In some transients, these two branches will operate identically, and so the use of multiplicity can cut the computing time nearly in half. The use of multiplicity is optional, but it is usually worthwhile to see if a particular transient can benefit from the use of multiplicity and to reconfigure the plant (and therefore the nodalization) so as to eliminate duplicate branches.

Remember, one branch is a duplicate of another only if both have the same physical configuration and they can be expected to operate identically throughout the transient. Only then can multiplicity be applied.

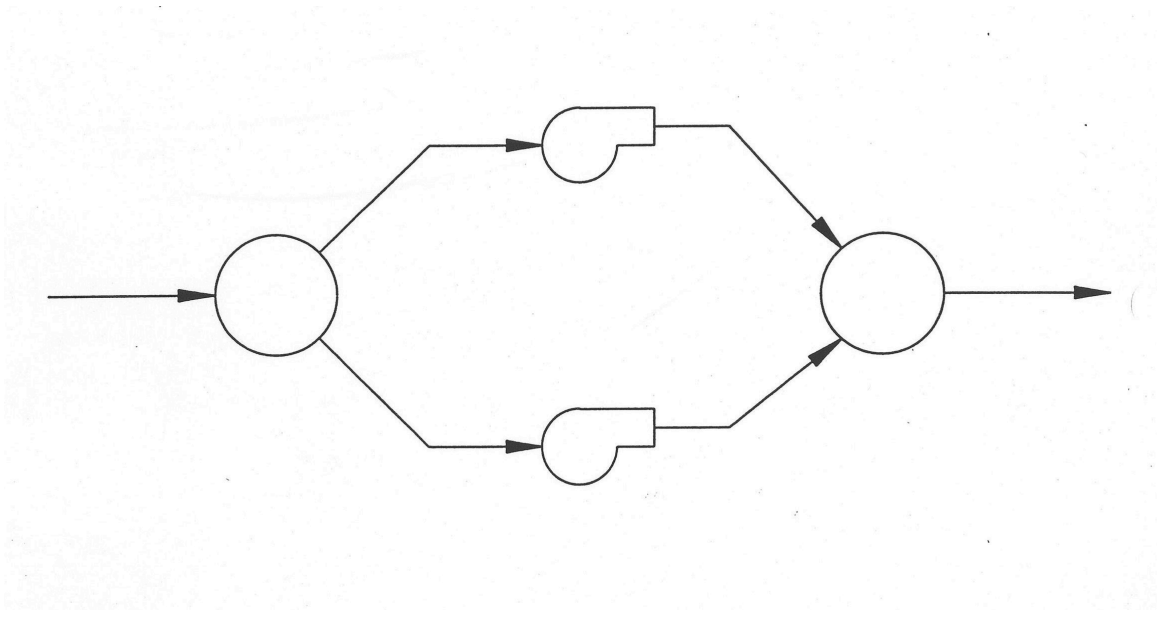


Figure 7.2.3: Exampled Problem for Demonstrating Multiplicity

7.3 Steam Generator Model

7.3.1 Once-Through Steam Generator

In an once-through steam generator subcooled feedwater enters the bottom on the water side and superheated steam comes out the top (see Figure 7.3.1). Considering that there is a transition boiling zone, the water side is naturally divided into three regions. Figure 7.3.2 shows a detailed schematic of the once-through steam generator. The top of the subcooled zone and the bottom of the boiling zone is defined by the point of saturated liquid enthalpy. The top of the boiling zone and the bottom of the superheated vapor zone is defined by the point of saturated vapor enthalpy. This is the situation during normal steady-state operation. Various transient conditions can produce any situation from a steam generator filled with subcooled water to total dry-out on the water side. The current model can calculate this whole spectrum of conditions with one proviso: there must always be a subcooled liquid region of some finite length at the inlet of the steam generator; but this length can be extremely small. Another way of stating this assumption is that there is no provision for two-phase fluid or superheated vapor in the inlet plenum of the steam generator. Going to the other end of the spectrum of

transient cases, the complete disappearance of both the boiling and superheated vapor zones can be calculated. If, however, there is a superheated vapor zone, there must, of course, be a boiling zone. Each of the three zones, therefore, is treated as a separate calculation with its own node structure and providing boundary conditions for the adjacent region or regions even as each zone length changes during the transient.

The steam generator is, of course, one module in the balance-of-plant sequence. As far as the system is concerned, it represents a pressure drop and a momentum source or sink from a hydrodynamic point of view. The balance-of-plant calculation produces pressures at the inlet and outlet plena of the steam generator as well as the mass flow into the steam generator. As will be shown later, the steam generator model itself calculates outlet flows from the steam generator. The steam generator provides the balance-of-plant momentum equation with an estimate of the pressure drop across the steam generator. This will be discussed later.

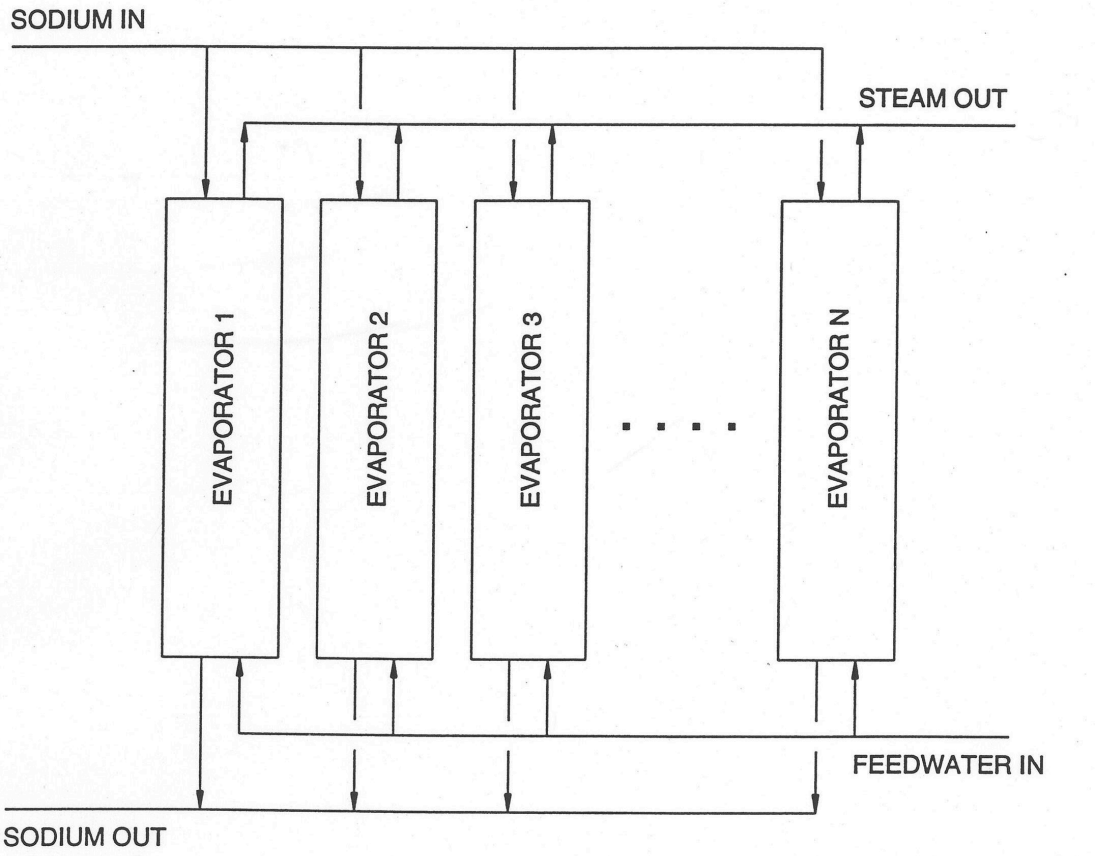


Figure 7.3.1: Once-Through Steam Generator

There are two main points to emphasize here, however. First, the inlet and outlet pressures provided by the balance-of-plant momentum equation are simply averaged and this average pressure is used at each time step by the steam generator to calculate properties. Thus no account is taken of the variation of pressure across the steam generator for the purpose of calculating properties. For many and probably most transient calculations of interest, this pressure variation is small and, although not trivial, can be safely neglected in the total context of the calculation. There are some transients, however, involving large pressure reductions downstream of the steam generator, which would produce a significant pressure variation across the steam generator, the neglect of which could lead to some level of inaccuracy.

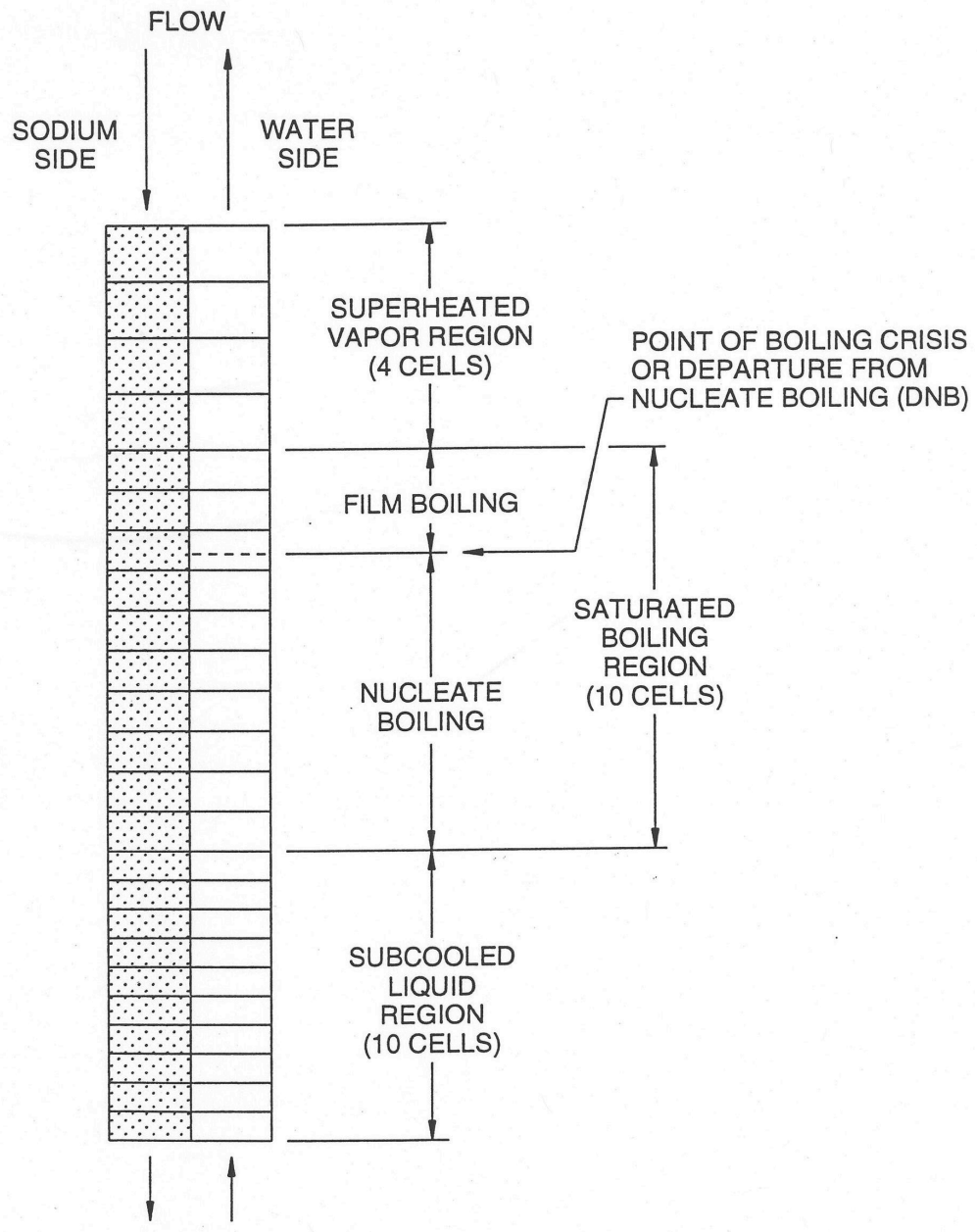


Figure 7.3.2: Schematic of Once-Through Steam Generator

The second point to emphasize is, given the estimate of the pressure drop across the steam generator, the inlet flow is provided as a boundary condition for the steam generator. There is no momentum equation coupled to the mass and energy conservation equations which characterize the regions in the steam generator to produce velocities. Instead the mass flows above the subcooled zone result from the mass and energy equations alone (as shown below), given the inlet flow as a driving function.

The subcooled liquid and superheated vapor zones each have their own heat transfer regime. The boiling zone has two heat transfer regimes separated at the boiling crisis or departure-from-nucleate-boiling (DNB) point. No smoothing or intermediate regimes are used between these four regimes. This provides of course calculation convenience. Heat transfer phenomena in a steam generator are much more complicated. The adequacy of this heat transfer scheme will be judged by benchmarking against experimental data. The DNB point is crucial to properly characterizing the boiling zone. But tracking the point produces calculation difficulties. This will be explained in some detail later. Suffice it to say here that the DNB point is assumed to be at the intersection of two curves, one representing the local heat flux at the tube wall and the other representing the heat flux required for the boiling crisis to occur. In this way, the point of maximum boiling heat flux is tracked. This DNB point is tracked continuously within the node structure of the boiling zone and the heat flux in the cell where the DNB point is a prorated average of the two boiling heat transfer regimes since the volumetric heat flux is always calculated on a cell-average basis. Depending on the fineness of the node structure, this produces some amount of inaccuracy and approximation to the real physical situation.

The following are general forms of the continuity and of the enthalpy form of the energy conservation equation in one dimension:

$$\frac{\partial \rho}{\partial t} = -\frac{\partial G}{\partial z} \tag{7.3-1}$$

$$\frac{\partial (\rho h)}{\partial t} = -\frac{\partial \left(Gh\right)}{\partial z} + Q - \left(\tau : \nabla v\right) + \frac{\partial P}{\partial t} + v\frac{\partial P}{\partial z} \tag{7.3-2}$$

Q is a volumetric heat source, $-\left(\tau : \nabla v\right)$ represents viscous dissipation and $v(\partial P/\partial z)$ is a work-energy conversion term (representing feedwater pump work, for example). The viscous dissipation term will be dropped for this application because it is small compared to other terms. The work term will also be neglected for the same reason although it is possible that in certain extreme conditions the term could be of some significance. The general energy equation thus becomes,

$$\frac{\partial (\rho h)}{\partial t} = -\frac{\partial \left(Gh\right)}{\partial z} + Q + \frac{\partial P}{\partial t} \tag{7.3-3}$$

Incompressible flow is assumed in the subcooled liquid zone. The balance-of-plant momentum equation provides the inlet mass flow, as stated above and thus the mass flow for the whole zone ($\Delta G = 0$). Therefore no continuity equation is needed to characterize the region. The enthalpy level and shape and the subcooled region length are all that need be solved for with a coupled set of nodal energy conservation equations. The boundary conditions for flow are the constant mass flow and for energy, the inlet enthalpy and the saturated liquid enthalpy unless the zone reaches the top of the steam generator in which case the region length is known and the outlet enthalpy is calculated. Since is assumed to be zero, the LHS of the energy equation, Eq. (7.3-3), can be simplified. The density changes over the transient as a result of changes in pressure and enthalpy but these changes are taken into account by updating the density at the end of each time step in the transient after new enthalpies and pressures are obtained.

In the boiling region, compressible flow is calculated with sets of nodal mass and energy conservation equations. The equations are formulated in terms of the void fraction instead of density and enthalpy. This is conveniently done since saturation conditions are always assumed and a no slip condition between phases is assumed; and also because saturation properties are functions of pressure alone. Thus simultaneous nodal continuity and energy equations are used to solve for mass flows and void fractions. The boundary conditions at the bottom of the zone are the single-phase liquid flow and the saturated liquid enthalpy (i.e. void fraction zero). At the top of the zone, there is either the saturated vapor enthalpy (i.e. void fraction 1.0) or, if the boiling zone extends all the way to the top of the steam generator, then the void fraction is a free variable and only lower boundary conditions are

required. When the zone does not extend to the top of the steam generator, then the upper boundary condition of the saturated vapor enthalpy is used by requiring that the length of the zone be adjusted until the upper boundary condition is satisfied.

Compressible flow is also assumed in the superheated vapor region. The boundary conditions at the bottom of the zone are the saturated vapor enthalpy and the mass flow calculated at the top of the boiling zone. Since the length of the zone is known, simultaneous nodal mass and energy equations are used to solve for enthalpies and mass flows all the way to the top of the steam generator.

On the sodium side, there is no change of phase and consequently the liquid can be adequately treated with incompressible flow. There is a calculation of the sodium flow external to the steam generator calculation so that, as far as the steam generator is concerned, the mass flow is given. Therefore no continuity equation is required. Besides the mass flow, the only boundary condition required is the inlet sodium temperature at the top of the steam generator. Only the nodal energy equation is required to solve for the enthalpy shape on the sodium side. In addition, given the relatively stable and low pressure conditions on the sodium side, the pressure term in the energy equation Eq. (7.3-3) can be safely neglected.

The heat capacity of the tube wall separating the water and the sodium must be taken into account. Therefore its effect on the heat transfer from the sodium to the water is treated by means of a wall temperature calculation. With no convective or pressure terms the energy equation Eq. (7.3-3) becomes much simplified. Also, the density is assumed to be temperature independent which further facilitates the solution.

7.3.2 Recirculation-Type Steam Generator

Figure 7.3.3 depicts a steam generator which consists of several evaporators, several superheaters, a steam drum and a recirculation loop. Subcooled water is pumped into the bottom of the evaporator. Within the evaporator the water boils and a two-phase fluid, typically of very high quality, leaves the top of the evaporator and enters the steam drum. One exit line from the steam drum transports saturated steam to the bottom of the superheater where more heat is added so that highly superheated steam leaves the top available for the turbines. The other exit line from the steam drum transports saturated liquid to the pump but before reaching the pump it is mixed with feedwater. This mixture is substantially subcooled, therefore, and is pumped back to the evaporator to complete the cycle.

The modeling of the evaporator can be done with the once-through steam generator model which is designed to model any situation from a liquid-filled steam generator to the normal operating condition for a once-through type with superheated vapor exiting the top. Thus the physical modeling assumptions of the previous section apply to the evaporator.

The modeling of the superheater is different, however, than the super-heated vapor zone of the previous section. Without elaborating on the details, it is sufficient here to say that the momentum equation of the balance-of-plant model is much more tractable if the assumption of incompressible flow is made for the superheater. Since the superheater operates at quite high pressures, this incompressibility assumption is probably adequate in most transient conditions. There may, however, be certain conditions when the pressure in the superheater is greatly reduced when inaccuracies may result from this assumption. A study of this effect would have to be undertaken to decide the issue and it has not been done so far. The lower boundary condition besides the given mass flow is the inlet enthalpy (i.e. the saturated vapor enthalpy). Thus the enthalpy shape is determined given these conditions. The density is updated each time step during the transient after new enthalpies and pressures have been determined.

The steam drum is modeled as a zero-dimensional reservoir in the sense that perfect mixing of all incoming fluid and the pre-existing separated two phases is presumed. There is one proviso here, however: the liquid level must be tracked so that appropriate action can be taken when liquid may enter the pipe to the superheater or vapor may enter the pipe to the recirculation pump.

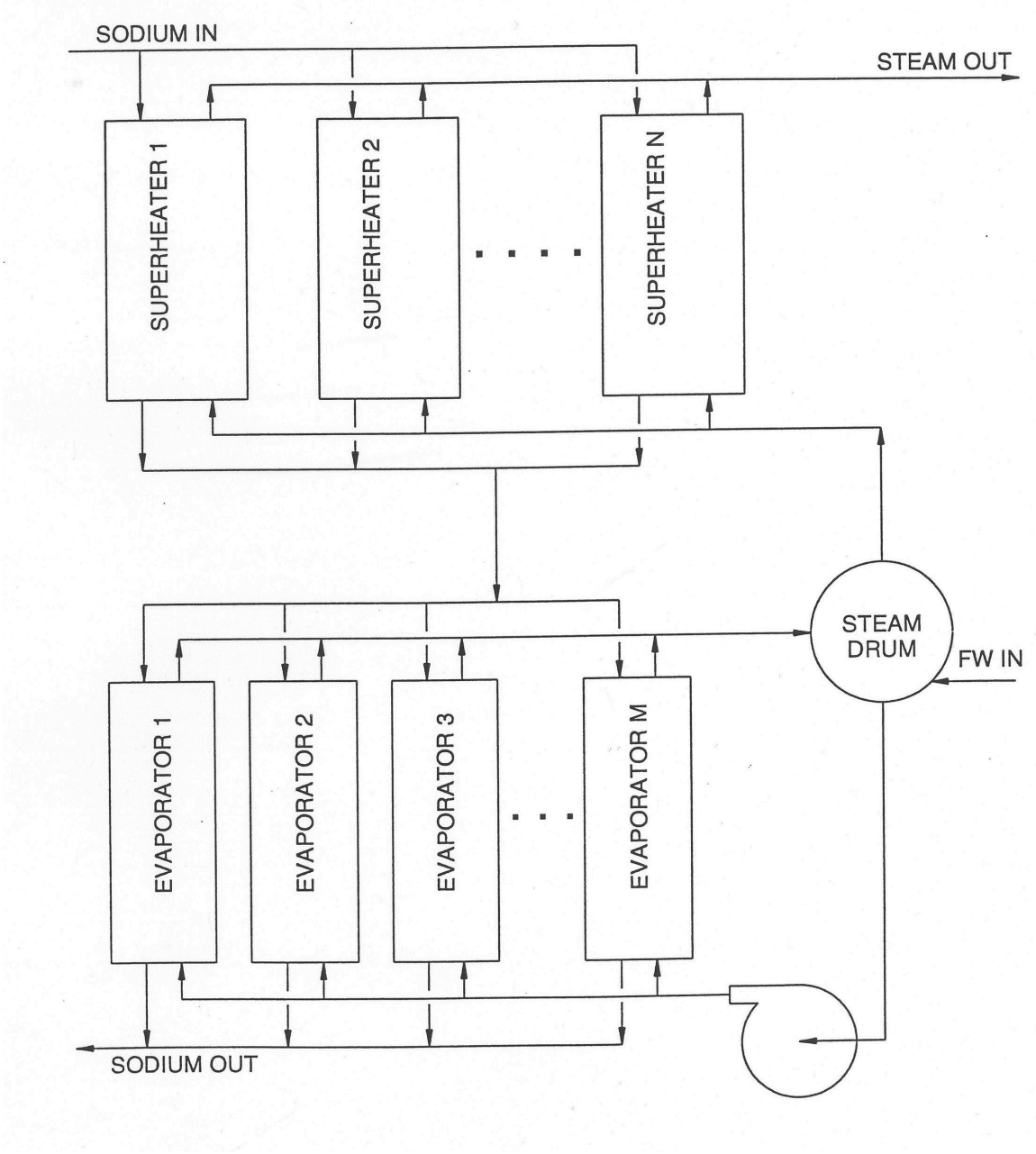


Figure 7.3.3: Steam Generator with Separate Evaporator and Superheaters and Recirculation Loop

7.3.3 Numerical Solution Methods

7.3.3.1 Once-Through Steam Generator

7.3.3.1.1 General Forms of the Conservation Equations

Before the individual solution methods for each water side region and the sodium side can be considered, general forms of the continuity and energy equations must be developed. Equations will be given for one node of the multi-node system of equations. Integration of the continuity Eq. (7.3-1) over the length of one cell from Z_i to Z_{i+1} gives,

$$\int_{Z_i}^{Z_{i+1}} \frac{\partial}{\partial t}\rho dz = -\int_{Z_i}^{Z_{i+1}} \frac{\partial}{\partial z}G dz \tag{7.3-4}$$

According to Leibnitz's Theorem,

$$\int_{Z_i}^{Z_{i+1}} \frac{\partial}{\partial z}\rho dz = \frac{d}{dz}\int_{Z_i}^{Z_{i+1}} \rho dz - \rho_{i+1}\dot{Z}_{i+1} + \rho_i \dot{Z}_i \tag{7.3-5}$$

Eq. (7.3-4) becomes,

$$\frac{d}{dt}\int_{Z_i}^{Z_{i+1}} \rho dz - \rho_{i+1}\dot{Z}_{i+1} + \rho_i \dot{Z}_i = -\int_{Z_i}^{Z_{i+1}} \frac{\partial}{\partial z}G dz \tag{7.3-6}$$

Therefore,

$$\frac{d}{dt}\left\{\overline{\rho}_i \Delta Z_i\right\} - \rho_{i+1}\dot{Z}_{i+1} + \rho_i \dot{Z}_i = -\Delta G_i \tag{7.3-7}$$

Donor-cell differencing is used to enhance numerical stability. In order to write the equation in donor-cell form, let ρ_{i+1} replace the average value over the interval, $\overline{\rho}_i$ and simplify,

$$\dot{\rho}_{i+1}\Delta Z_i - \Delta\rho_i \dot{Z}_i = -\Delta G_i \tag{7.3-8}$$

Integration of the enthalpy form of the energy Eq. (7.3-3) from Z_i to Z_{i+1} gives,

$$\int_{Z_i}^{Z_{i+1}} \frac{\partial}{\partial t}(\rho h)\, dz = -\int_{Z_i}^{Z_{i+1}} \frac{\partial}{\partial z}(Gh)\, dz + \int_{Z_i}^{Z_{i+1}} Q dz + \int_{Z_i}^{Z_{i+1}} \frac{\partial}{\partial z}P dz \tag{7.3-9}$$

According to Leibnitz's theorem,

$$\int_{Z_i}^{Z_{i+1}} \frac{\partial}{\partial t}(\rho h) dz = \frac{d}{dt}\int_{Z_i}^{Z_{i+1}} (\rho h) dz - (\rho h)_{i+1}\dot{Z}_{i+1} + (\rho h)_i \dot{Z}_i \tag{7.3-10}$$

$$\int_{Z_i}^{Z_{i+1}} \frac{\partial}{\partial t}P dz = \frac{d}{dt}\int_{Z_i}^{Z_{i+1}} P dz - P_{i+1}\dot{Z}_{i+1} + P_i \dot{Z}_i \tag{7.3-11}$$

Eq. (7.3-9) becomes

$$\begin{aligned}\frac{d}{dh}\int_{Z_i}^{Z_{i+1}} (\rho h)\, dz - (\rho h)_{i+1}\dot{Z}_{i+1} + (\rho h)_i \dot{Z}_i = &-\int_{Z_i}^{Z_{i+1}} \frac{\partial}{\partial z}(Gh)\, dz + \int_{Z_i}^{Z_{i+1}} Q dz \\ &+\frac{d}{dt}\int_{Z_i}^{Z_{i+1}} P dz - P_{i+1}\dot{Z}_{i+1} + P_i \dot{Z}_i\end{aligned} \tag{7.3-12}$$

Therefore,

$$\frac{d}{dt}\left\{\left(\overline{\rho h}\right)_i \Delta Z_i\right\} - (\rho h)_{i+1}\dot{Z}_{i+1} + (\rho h_i)\dot{Z}_i = -\Delta\left(Gh\right)_i + Q_i\Delta Z_i + \frac{d}{dt}\left\{\overline{P}_i\Delta Z_i\right\} - P_{i+1}\dot{Z}_{i+1} + P_i\dot{Z}_i \tag{7.3-13}$$

In order to write Eq. (7.3-13) in donor-cell form, let $(\rho h)_{i+1}$ replace the average value over the interval $[\rho h]_i$ let P_{i+1} replace $\overline{P}_i$, and simplify,

$$\left(\rho\dot{h}\right)_{i+1}\Delta Z_i - \Delta\left(\rho h\right)_i\dot{Z}_i = -\Delta\left(Gh\right)_i + Q_i\Delta Z_i + \dot{P}_{i+1}\Delta Z_i - \Delta P_i\dot{Z}_i \tag{7.3-14}$$

As discussed before, the pressure variation across the steam generator is neglected, i.e., $\Delta P_i = 0$, and Eq. (7.3-14) becomes

$$\left(\rho\dot{h}\right)_{i+1}\Delta Z_i - \Delta\left(\rho h\right)_i\dot{Z}_i = -\Delta\left(Gh\right)_i + Q_i\Delta Z_i + \dot{P}\Delta Z_i \tag{7.3-15}$$

Each of the three regions on the water side is divided into a fixed number of cells. Since the region lengths vary during the transient, the cell lengths also vary and are thus a constant fraction of the varying region length. Thus $\Delta Z_i = \frac{1}{n}Z_x$ where Z_x is the current length of region x and n is the number of cells in region x. Therefore the subscript i can be dropped for ΔZ_i in equations for a given region. On the sodium side, the same node structure is used as on the water side in order to simplify the calculation, although the precise node structure on the sodium side is not nearly as important as on the water side since there is no change of phase and properties thus calculated parameters change gradually and smoothly over the length of the steam generator.

All spatially variable parameters except two are evaluated at the cell boundary so that there are $n + 1$ values needed to characterize a region, where n is the number of cells in the region. The two exceptions are the tube wall enthalpy and the heat flux. The volumetric heat source is most conveniently calculated on a cell-average basis so that n cells exactly encompass the whole of the heat transfer for a given region. If the heat source were calculated at the cell edge, then half-cells would have to be used at the ends of a region where the heat transfer coefficients change form. In order to calculate the temperature gradients for the heat flux, linear averages over the cell length are computed from the cell-edge values for water and sodium. However, there is no need to calculate the tube wall temperature at the cell edge and a cell-centered value is most convenient for the heat flux calculation. Other parameters such as enthalpies, void fractions, mass fluxes, etc. are most conveniently calculated at the cell-edge since this is where the boundary conditions are defined.

7.3.3.1.2 Subcooled Liquid Region

In the subcooled region, the mass flow is assumed to be uniform throughout the zone due to incompressible flow. Each time step during the transient an updated inlet mass flow is provided to the steam generator from the explicitly-coupled momentum equation. Therefore no continuity equation is required. The coupled set of nodal energy conservation equations are used to determine the length of the zone and the nodal enthalpies simultaneously using the inlet enthalpy and the saturated liquid enthalpy as boundary conditions. Or, alternatively, when liquid fills the steam generator and the zone length is known, the outlet enthalpy is instead determined.

Using Eq. (7.3-15) and setting $\dot{\rho} = 0$ because of the incompressible flow, and recalling that ΔZ_i is invarient within a zone, the following results for node i,

$$\dot{h}_{i+1}\rho_{i+1}\Delta Z - \Delta\left(\rho h\right)_i\dot{Z}_i = -\Delta\left(Gh\right)_i + Q_i\Delta Z + \dot{P}\Delta Z \tag{7.3-16}$$

The following is Eq. (7.3-16) in finite difference form,

$$h_{i+1}^{k+1}\frac{1}{\Delta t}\rho_{i+1}^k\frac{1}{n}Z_{SC}^k - h_{i+1}^k\frac{1}{\Delta t}\rho_{i+1}^k\frac{1}{n}Z_{SC}^{k+1} - \overline{\rho}_i^k\left(h_{i+1}^k - h_i^k\right)\frac{i-1}{n} \times\frac{1}{\Delta t}\left(Z_{SC}^{k+1} - Z_{SC}^k\right) = -G^{k+1}\left(h_{i+1}^{k+1} - h_i^{k+1}\right) + Q_i^k\frac{1}{n}Z_{SC}^{k+1} + \dot{P}\frac{1}{n}Z_{SC}^{k+1} \tag{7.3-17}$$

There are several points to note concerning Eq. (7.3-17). The ΔZ in the first term on the LHS is treated semi-implicitly in time since h_{i+1}^{k+1} is multiplied by Z_{SC}^{k}, the value at the beginning of the time step and h_{i+1}^{k+1} is multiplied by Z_{SC}^{k+1}, the value at the end-of-step which needs to be determined. h_{i+1}^{k+1} cannot be multiplied by the end-of-step value without making the equation set non-linear. Making the term semi-implicit as opposed to fully explicit has been found to enhance stability in the calculation. The next point is that the density is entirely explicit in time. It is updated at the end of every time step as enthalpy and pressure change. This is consistent with the assumption of incompressibility as density changes slowly and gradually over time. Note also the use of $\overline{\rho}_i^k$ in the third term on the LHS of Eq. (7.3-17). $\Delta\rho h_i$ becomes $\rho\Delta h_i$ in order to preserve the proper sign of the term. Using $\overline{\rho}_i^k$ which is $0.5\left(\rho_{i+1}^k + \rho_i^k\right)$, introduces only a small error so long as the mesh structure is not too coarse. Δh_i must be explicit in time because it is multiplied by Z_{SC}^{k+1} and the equation must be kept linear. G^{k+1} in the convective term implies that the inlet mass flow is updated before the steam generator solution begins. The volumetric heat source is totally explicit since the temperatures used in calculating the ΔT and the properties contained in the heat transfer correlation (see Section 7.7.2) cannot be made implicit without making the equation insoluble.

Rearranging Eq. (7.3-17) according to coefficients there are n equations of the following form,

$$\begin{aligned}\left[-G^{k+1}\right] h_i^{k+1} + \left[\frac{1}{\Delta t}\rho_{i+1}^k \frac{1}{n} Z_{SC}^k + G^{k+1}\right] h_{i+1}^{k+1} \\ + \left[-h_{i+1}^k \frac{1}{\Delta t}\rho_{i+1}^k \frac{1}{n} - \overline{\rho}_i^k \left(h_{i+1}^k - h_i^k\right) \frac{i-1}{n}\frac{1}{\Delta t} - Q_i^k \frac{1}{n} - \dot{P}\frac{1}{n}\right] Z_{SC}^{k+1} \\ = -\overline{\rho}_i^k \left(h_{i+1}^k - h_i^k\right) \frac{i-1}{n}\frac{1}{\Delta t} Z_{SC}^{k+1}\end{aligned} \tag{7.3-18}$$

Eq. (7.3-18) can be written as,

$$a_i h_i + b_i h_{i+1} + c_i Z_{SC} = d_i \tag{7.3-19}$$

In the case where the liquid region does not reach the top of the steam generator, h_1 and h_{n+1} are known, since they are the inlet enthalpy and $h_f . h_2 - h_n$ and Z_{SC} are unknown (n unknowns) and are solved for as follows. (The superscript $k+1$ is dropped as unnecessary.)

From the first equation of the equation set Eq. (7.3-19), solve for h_2, then for h_3 from the second equation and so on,

$$\begin{aligned} h_2 &= e_1 + f_1 Z_{SC}; \quad e_1 = \frac{d_1 - a_1 h_1}{b_1}, \quad f_1 = -\frac{c_1}{b_1} \\ h_3 &= e_2 + f_2 Z_{SC}; \quad e_2 = \frac{d_2 - a_2 e_1}{b_2}, \quad f_2 = -\frac{c_2 + a_2 f_1}{b_2} \\ h_{i+1} &= e_i + f_i Z_{SC}; \quad e_i = \frac{d_i - a_i e_{i-1}}{b_i}, \quad f_i = -\frac{c_i + a_i f_{i-1}}{b_i} \end{aligned} \tag{7.3-20}$$

Finally, in the equation for h_{n+1}, Z_{SC} can be solved for since h_{n+1} is known. Then each of $h_2 - h_n$ can be solved for since they are all functions of only Z_{SC} in the equation set Eq. (7.3-20).

If the steam generator is filled with liquid, then the outlet enthalpy is unknown and there are $n`sequationswith : math : `h_2 - h_{n+1}$ as the unknowns. Eq. (7.3-16) becomes the following in finite difference form (the zone length, now constant as the length of the steam generator, is denoted simply as Z_{SC}),

$$\frac{1}{\Delta t}\rho_{i+1}^k \frac{1}{n} Z_{SC}\left(h_{i+1}^{k+1} - h_{i+1}^k\right) = -G^{k+1}\left(h_{i+1}^{k+1} - h_i^{k+1}\right) + \frac{1}{n} Z_{SC}\left(Q_i^k + \dot{P}\right) \tag{7.3-21}$$

The following equations are simply solved from the bottom to the top of the steam generator successively,

$$h_{i+1}^{k+1} = \frac{\frac{1}{\Delta t} h_{i+1}^k \rho_{i+1}^k \frac{1}{n} Z_{SC} + G^{k+1} h_i^{k+1} + Q_i \frac{1}{n} Z_{SC} + \dot{P}\frac{1}{n} Z_{SC}}{\frac{1}{\Delta t}\rho_{i+1}^k \frac{1}{n} Z_{SC} + G^{k+1}} \tag{7.3-22}$$

7.3.3.1.3 Boiling Region

In the boiling zone, the fluid is treated as compressible. Simultaneous nodal equations are solved for void fraction, mass flow and region length. Boundary conditions are the saturated liquid enthalpy and subcooled region mass flow at the bottom of the boiling zone and either the saturated vapor enthalpy at the top of the zone or, if there is no superheated vapor zone, the region length is defined and the outlet enthalpy is determined. Only the pressure and the volumetric heat source are treated explicitly in time. The void fraction, the mass flow and the region length are all treated in fully implicit fashion. An iterative method is used to solve for the region length. The current value of the region length is held constant for each pass in the iteration while nodal void fractions and mass flows are calculated from the mass and energy equations. When the uppermost void fraction in the boiling zone is computed at the end of an iteration, its value is compared to 1.0 and the region length is adjusted appropriately and the iterative process continues until convergence. When the boiling zone extends to the top of the steam generator, the same method is used but there is no iteration since the region length is known.

First, a number of definitions and identities concerning a two-phase fluid must be reviewed. ρ_f and h_f are the saturated liquid density and enthalpy respectively. ρ_g and h_g are the saturated vapor density and enthalpy. $\rho_{fg} = \rho_g - \rho_f . h_{fg} = h_g - h_f$ The density of the two-phase mixture is, with α denoting the void fraction,

$$\rho = \alpha\rho_{fg} + \rho_f \tag{7.3-23}$$

The enthalpy of the two-phase mixture is, with χ denoting the quality,

$$h = \chi h_{fg} + h_f \tag{7.3-24}$$

Since homogeneous flow is assumed in the two-phase zone,

$$\chi = \frac{\alpha\rho_g}{\rho} = \frac{\alpha\rho_g}{\rho_f + \alpha\rho_{fg}} \tag{7.3-25}$$

Defining $(h\rho)_{fg}$ as $h_g\rho_g - h_f\rho_f$, Eq. (7.3-23), Eq. (7.3-24), and Eq. (7.3-25) imply,

$$(h\rho) = h_f\rho_f + \alpha\,(h\rho)_{fg} \tag{7.3-26}$$

The general continuity Eq. (7.3-8) is written in terms of the nodal void fraction, α,

$$\left(\dot{\alpha}_{i+1}\rho_{fg} + \alpha_{i+1}\dot{\rho}_{fg} + \dot{\rho}_f\right)\Delta Z - \left(\alpha_{i+1} - \alpha_i\right)\rho_{fg}\dot{Z}_i = -\Delta G_i \tag{7.3-27}$$

The finite difference form of Eq. (7.3-27) is,

$$\begin{aligned}\left[\frac{1}{\Delta t}\left(\alpha_{i+1}^{k+1} - \alpha_{i+1}^{k}\right)\rho_{fg}^{k} + \alpha_{i+1}^{k+1}\dot{\rho}_{fg}^{k} + \dot{\rho}_f^{k}\right]\frac{1}{n}Z_{TP}^{k+1} \\ - \left(\alpha_{i+1}^{k+1} - \alpha_i^{k+1}\right)\rho_{fg}^{k}\left[\frac{i-j}{n}\frac{1}{\Delta t}\left(Z_{TP}^{k+1} - Z_{TP}^{k}\right) + \dot{Z}_{SC}\right] = -\left(G_{i+1}^{k+1} - G_i^{k+1}\right)\end{aligned} \tag{7.3-28}$$

Rearranging Eq. (7.3-28) according to coefficients of α_{i+1}^{k+1} and G_{i+1}^{k+1} the following results,

$$\begin{aligned}\left\{\frac{1}{\Delta t}\rho_{fg}^{k}\frac{1}{n}Z_{TP}^{k+1} + \dot{\rho}_{fg}^{k}\frac{1}{n}Z_{TP}^{k+1} - \rho_{fg}^{k}\left[\frac{i-j}{n}\frac{1}{\Delta t}\left(Z_{TP}^{k+1} - Z_{TP}^{k}\right) + \dot{Z}_{SC}\right]\right\}\alpha_{i+1}^{k+1} \\ + G_{i+1}^{k+1} + \left\{-\frac{1}{\Delta t}\alpha_{i+1}^{k}\rho_{fg}^{k}\frac{1}{n}Z_{TP}^{k+1} + \dot{\rho}_f^{k}\frac{1}{n}Z_{TP}^{k+1} + \alpha_i^{k+1}\rho_{fg}^{k}\left[\frac{i-j}{n}\frac{1}{\Delta t}\right.\right. \\ \left.\left.\times\left(Z_{TP}^{k+1} - Z_{TP}^{k}\right) + \dot{Z}_{SC}\right] - G_i^{k+1}\right\} = 0\end{aligned} \tag{7.3-29}$$

Eq. (7.3-29) can be written as,

$$a \times \alpha_{i+1}^{k+1} + G_{i+1}^{k+1} + c = 0 \tag{7.3-30}$$

Writing the general energy Eq. (7.3-15) in terms of α results in the following,

$$\begin{aligned}\left[\dot{\alpha}_{i+1}\left(h\rho\right)_{fg}+\alpha_{i+1}\left(h\dot{\rho}\right)_{fg}+\left(h\dot{\rho}\right)_{f}\right]\Delta Z-\left(\alpha_{i+1}-\alpha_{i}\right)\left(h\rho\right)_{fg}\dot{Z}_{i}\\ =-\Delta\left(Gh\right)_{i}+Q_{i}\Delta Z+\dot{P}\Delta Z\end{aligned} \tag{7.3-31}$$

The finite difference form of Eq. (7.3-31) is,

$$\begin{aligned}&\left[\frac{1}{\Delta t}\left(\alpha_{i+1}^{k+1}-\alpha_{i+1}^{k}\right)\left(h\rho\right)_{fg}^{k}+\alpha_{i+1}^{k+1}\left(h\dot{\rho}\right)_{fg}^{k}+\left(h\dot{\rho}\right)_{f}^{k}\right]\frac{1}{n}Z_{TP}^{k+1}-\left(\alpha_{i+1}^{k+1}-\alpha_{i}^{k+1}\right)\\ &\left(h\rho\right)_{fg}^{k}\left[\frac{i-j}{n}\frac{1}{\Delta t}\left(Z_{TP}^{k+1}-Z_{TP}^{k}\right)+\dot{Z}_{SC}\right]=-\left[G_{i+1}^{k+1}\left(h_{f}^{k}+\frac{\alpha_{i+1}^{k+1}\rho_{g}^{k}h_{fg}^{k}}{\rho_{f}^{k}+\alpha_{i+1}^{k+1}\rho_{fg}^{k}}\right)+\right.\\ &\left.G_{i}^{k+1}\left(h_{f}^{k}+\frac{\alpha_{i}^{k+1}\rho_{g}^{k}h_{fg}^{k}}{\rho_{f}^{k}+\alpha_{i}^{k+1}\rho_{fg}^{k}}\right)\right]+Q_{i}^{k}\frac{1}{n}Z_{TP}^{k+1}+\dot{P}\frac{1}{n}Z_{TP}^{k+1}\end{aligned} \tag{7.3-32}$$

Rearranging Eq. (7.3-32) according to coefficients of α_{i+1}^{k+1} and G_{i+1}^{k+1}, the following results,

$$\begin{aligned}&\left\{\frac{1}{\Delta t}\left(h\rho\right)_{fg}^{k}\frac{1}{n}Z_{TP}^{k+1}+\left(h\dot{\rho}\right)_{fg}^{k}\frac{1}{n}Z_{TP}^{k+1}-\left(h\rho\right)_{fg}^{k}\left[\frac{i-j}{n}\frac{1}{\Delta t}\left(Z_{TP}^{k+1}-Z_{TP}^{k}\right)+\dot{Z}_{SC}\right]\right\}\\ &\times\alpha_{i+1}^{k+1}+G_{i+1}^{k+1}\left(h_{f}^{k}+\frac{\alpha_{i+1}^{k+1}\rho_{g}^{k}h_{fg}^{k}}{\rho_{f}^{k}+\alpha_{i+1}^{k+1}\rho_{fg}^{k}}\right)+\left\{-\alpha_{i+1}^{k}\frac{1}{\Delta t}\left(h\rho\right)_{fg}^{k}\frac{1}{n}Z_{TP}^{k+1}\right.\\ &+\left(h\dot{\rho}\right)_{f}^{k}\frac{1}{n}Z_{TP}^{k+1}+\alpha_{i}^{k+1}\left(h\rho\right)_{fg}^{k}\left[\frac{i-j}{n}\frac{1}{\Delta t}\left(Z_{TP}^{k+1}-Z_{TP}^{k}\right)+\dot{Z}_{SC}\right]\\ &\left.-G_{i}^{k+1}\left(h_{f}^{k}+\frac{\alpha_{i}^{k+1}\rho_{f}^{k}h_{fg}^{k}}{\rho_{f}^{k}+\alpha_{i}^{k+1}\rho_{fg}^{k}}\right)-Q_{i}^{k}\frac{1}{n}Z_{TP}^{k+1}-\dot{P}\frac{1}{n}Z_{TP}^{k+1}\right\}=0\end{aligned} \tag{7.3-33}$$

If a' and c' represent the coefficient of a_{i+1}^{k+1} and the constant term respectively, Eq. (7.3-33) becomes,

$$a'\alpha_{i+1}^{k+1}+G_{i+1}^{k+1}\left(h_{f}^{k}+\frac{\alpha_{i+1}^{k+1}\rho_{g}^{k}h_{fg}^{k}}{\rho_{f}^{k}+\alpha_{i+1}^{k+1}\rho_{fg}^{k}}\right)+c'=0 \tag{7.3-34}$$

When the mass Eq. (7.3-30) is substituted into Eq. (7.3-34), a quadratic in a_{i+1}^{k+1} results,

$$\begin{aligned}\left(\alpha_{i+1}^{k+1}\right)^{2}\left[a'\rho_{fg}^{k}-ah_{f}^{k}\rho_{fg}^{k}-a\rho_{g}^{k}h_{fg}^{k}\right]+\alpha_{i+1}^{k+1}\left[a'\rho_{f}^{k}-c\rho_{fg}^{k}h_{f}^{k}\right.\\ \left.-c\rho_{g}^{k}h_{fg}^{k}+c'\rho_{fg}^{k}-ah_{f}^{k}\rho_{f}^{k}+\left[c'\rho_{f}^{k}-ch_{f}^{k}\rho_{f}^{k}\right]=0\right.\end{aligned} \tag{7.3-35}$$

Several points need to be noted here. An inspection of Eq. (7.3-28) and Eq. (7.3-32) shows that both α_{i+1} and G_{i+1} are totally implicit in time. It cannot be emphasized enough how much this feature of the boiling zone numerical solution enhances the stability of the calculation compared to other numerical schemes which are semi-implicit in time which were also tried. In order to preserve a linearized set of equations, it is necessary that the void fractions and mass flows be semi-implicit in time and this has a strong tendency to produce instabilities. The only quantities which are explicit in time are the saturation properties which are functions of pressure alone and the volumetric heat source term. Saturation properties are very well behaved functions of time since pressure tends to be a relatively stable function of time in most transients and the saturation properties are not as sensitive to changes in pressure as other quantities are sensitive to changes over time.

The last point concerns the $k+1$ superscript on Z_{TP}, the length of the two-phase zone. What this indicates, as noted above, is that the current value of the zone length in the iterative process is used in Eq. (7.3-30) and Eq. (7.3-35). Eq. (7.3-25) is solved for α_{i+1}^{k+1} and then G_{i+1}^{k+1} is obtained from the continuity Eq. (7.3-30) for each node. At first, the value of Z_{TP} from the last time step (i.e. Z_{TP}^{k}) is used to solve for α_{i+1}^{k+1} and G_{i+1}^{k+1} over

the whole mesh starting at the bottom ($i = 1$) where a_i^{k+1} and a_i^{k+1} $(= 0)$ are the known boundary conditions. The solution then proceeds upwards until α_{i+1}^{k+1} is calculated. α_{i+1}^{k+1} compared to 1.0 and if it is greater than 1.0, Z_{TP} is reduced for the next iteration and if it is less than 1.0, Z_{TP} is increased. The search on Z_{TP} continues until α_{i+1}^{k+1} is sufficiently close to 1.0. If the boiling zone extends to the top of the steam generator, the same procedure is used, but no iteration on Z_{TP} is necessary since Z_{TP} is a fixed, known value.

7.3.3.1.4 Superheated Vapor Region

A compressible treatment of the vapor is used above the boiling zone and simultaneous nodal mass and energy equations are solved for nodal enthalpies and mass flows since the region length is known, being the remainder of the steam generator length after computing new subcooled and boiling zone lengths. The nodal densities and enthalpies are treated partially explicitly in time. Boundary conditions are the saturated vapor enthalpy and the mass flow at the bottom of the zone. The solution proceeds upwards to the top of the steam generator.

Since there is an expression for ρ as a function of enthalpy and pressure,

$$\dot{\rho} = \frac{\partial \rho}{\partial h}\frac{\partial h}{\partial t} + \frac{\partial \rho}{\partial P}\frac{\partial P}{\partial t} \tag{7.3-36}$$

By substituting Eq. (7.3-36) into the mass Eq. (7.3-8), an expression for G_{i+1} as a function of h_{i+1} results,

$$G_{i+1} = G_i - \left[\frac{\partial \rho}{\partial h}\dot{h}_{i+1} + \frac{\partial \rho}{\partial P}\dot{P}\right]\Delta Z + \Delta\rho_i \dot{Z}_i \tag{7.3-37}$$

Next, Eq. (7.3-36) is substituted into Eq. (7.3-15), the following results

$$\begin{aligned}\left[\frac{\partial \rho}{\partial h}\dot{h}_{i+1} + \frac{\partial \rho}{\partial P}\dot{P}\right] h_{i+1}\Delta Z + \rho_{i+1}\dot{h}_{i+1}\Delta Z - \Delta\left(\rho h\right)_i \dot{Z}_i \\ = -G_{i+1}h_{i+1} + G_i h_i + Q_i \Delta Z + \dot{P}\Delta Z\end{aligned} \tag{7.3-38}$$

Now, substituting the expression for G_{i+1} which results from the continuity Eq. (7.3-37) into the energy Eq. (7.3-38) and simplifying,

$$\rho_{i+1}\dot{h}_{i+1}\Delta Z + \rho_i h_i \dot{Z}_i = -G_i h_{i+1} + \rho_i h_{i+1}\dot{Z}_i + G_i h_i + Q_i \Delta Z + \dot{P}\Delta Z \tag{7.3-39}$$

The finite difference form of Eq. (7.3-39) is,

$$\begin{aligned}\frac{1}{\Delta t}\left(h_{i+1}^{k+1} - h_{i+1}^{k}\right)\rho_{i+1}^{k}\frac{1}{n}Z_{SH}^{k+1} + h_i^{k+1}\rho_i^{k+1}\frac{i-j}{n}\dot{Z}_{SH}^{k+1} = -G_i^{k+1}h_{i+1}^{k+1} + h_{i+1}^{k+1}\rho_i^{k+1}\frac{i-j}{n}\dot{Z}_{SH}^{k+1} \\ + G_{i+}^{k+1}h_i^{k+1} + Q_i^k\frac{1}{n}Z_{SH}^{k+1} + \dot{P}\frac{1}{n}Z_{SH}^{k+1}\end{aligned} \tag{7.3-40}$$

Eq. (7.3-40) is entirely implicit in enthalpy, mass flow and zone length. However, it is necessary to use the beginning-of-step value of ρ_{i+1} since density is a complicated function of enthalpy and pressure and there is no way to incorporate this function in Eq. (7.3-20) and preserve linearity. ρ_i, however, is entirely implicit since the solution proceeds upwards in the mesh and ρ_i can be updated as the solution proceeds along the mesh. By solving for h_{i+1}^{k+1}, the following results,

$$h_{i+1}^{k+1} = \frac{h_i^{k+1}\left(G_i^{k+1} - \rho_i^{k+1}\frac{i-j}{n}\dot{Z}_{SH}^{k+1}\right) + \frac{1}{n}Z_{SH}^{k+1}\left(\frac{1}{\Delta t}h_{i+1}^{k}\rho_{i+1}^{k} + Q_i^k + \dot{P}^k\right)}{G_i^{k+1} + \frac{1}{\Delta t}\rho_{i+1}^{k}\frac{1}{n}Z_{SH}^{k+1} - \rho_i^{k+1}\frac{i-j}{n}\dot{Z}_{SH}^{k+1}} \tag{7.3-41}$$

The finite-difference form of Eq. (7.3-37) is,

$$G_{i+1}^{k+1} = G_i^{k+1} - \left[\frac{\partial \rho}{\partial h}\frac{\left(h_{i+1}^{k+1} - h_{i+1}^{k}\right)}{\Delta t} + \frac{\partial \rho}{\partial P}\dot{P}\right]\frac{1}{n}Z_{SH}^{k+1} + \left(\rho_{i+1}^{k+1} - \rho_i^{k+1}\right)\dot{Z}_{SH} \tag{7.3-42}$$

After obtaining h_{i+1}^{k+1} from Eq. (7.3-41), ρ_{i+1}^{k+1} is calculated as a function of h_{i+1}^{k+1} and P^{k+1} and these are used to calculate G_{i+1}^{k+1} in Eq. (7.3-42). It should be noted that the partial derivatives of density with respect to enthalpy and pressure are evaluated with the h_{i+1}^{k+1} and ρ_{i+1}^{k+1} just calculated. Starting at the bottom of the mesh, G_1^{k+1} and h_1^{k+1} $(= h_g)$ are known and each parameter is solved for at the top of the cell at the $i+1$ location according to the above procedure up to the top of the steam generator.

7.3.3.1.5 Sodium Side Calculation

Incompressible flow is assumed on the sodium side because it is always in the liquid phase. Therefore there is no continuity equation. Also, since the sodium side is at relatively low pressure and has a stable pressure history, the pressure terms in the energy Eq. (7.3-13) are neglected. Next, since the sodium flows downward, in order to donor-cell the energy equation, $\left(\overline{\rho h}\right)_i$ is set equal to $(\rho h)_i$. Lastly, it is convenient to assume that G is positive for downward-flowing sodium which means that $-\Delta\left(Gh\right)_i$ becomes $-G\left[h_i - h_{i+1}\right]$. Thus Eq. (7.3-13) becomes,

$$\left(\rho\dot{h}\right)_i \Delta Z - \left[(\rho h)_{i+1} - (\rho h)_i\right] \dot{Z}_{i+1} = -G\left(h_i - h_{i+1}\right) + Q_i \Delta Z \tag{7.3-43}$$

Assuming $\dot{\rho} = 0$ because of incompressible flow and rewriting Eq. (7.3-43) in terms of temperature (and neglecting the $\dot{c}_p$ term as unimportant), the following results,

$$\overline{\rho}_i \overline{c}_{p,i} \dot{T}_i \Delta Z - \overline{\rho}_i \overline{c}_{p,i} \left(T_{i+1} - T_i\right) \dot{Z}_{i+1} = -G \overline{c}_{p,i} \left(T_i - T_{i+1}\right) + Q_i \Delta Z \tag{7.3-44}$$

The $\overline{c}_{p,i}$ and the $\overline{\rho}_i$ are computed using the sodium temperature at the cell-center. (This is a slight inaccuracy but, given the fact that liquid sodium properties are so well-behaved and change so gradually, it is of small consequence.) The finite difference form of Eq. (7.3-44) is,

$$\frac{\overline{\rho}_i^k}{\Delta t}\left(T_i^{k+1} - T_i^k\right) \Delta Z^{k+1} - \overline{\rho}_i^k \left(T_{i+1}^{k+1} - T_i^k\right) \dot{Z}_{i+1}^{k+1} = -G^{k+1}\left(T_i^{k+1} - T_{i+1}^{k+1}\right) + \frac{\Delta Z^{k+1}}{\overline{c}_{p,i}} Q_i^k \tag{7.3-45}$$

Note that the $\overline{\rho}_i$ and the $\overline{c}_{p,i}$ are computed with beginning-of-step temperatures and that the ΔZ and $\dot{Z}_{i+1}$ are the end-of-step quantities just calculated in the water side calculation. Q_i is, of course, explicit in time as usual and the mass flow provided by the external sodium loop calculation is the new end-of-time-step value. If Eq. (7.3-45) is now solved for T_i^{k+1}, the following results,

$$T_i^{k+1} = \frac{T_{i+1}^{k+1}\left(G^{k+1} + \overline{\rho}_i^k \dot{Z}_{i+1}^{k+1}\right) + T_i^k \frac{\Delta Z^{k+1}}{\Delta t}\overline{\rho}_i^k + \frac{\Delta Z^{k+1}}{\overline{c}_{p,i}} Q_i^k}{\frac{\Delta Z^{k+1}}{\Delta t}\overline{\rho}_i^k + \overline{\rho}_i^k \dot{Z}_{i+1}^{k+1} + G^{k+1}} \tag{7.3-46}$$

Starting at the top of the steam generator, with the new inlet sodium temperatures at the end of the time step, the calculation proceeds downward to the bottom of the mesh.

7.3.3.1.6 Wall Temperature Calculation

The heat capacity of the tube wall must be taken into account during the transient. Since there is no convective term in the energy equation, central differencing is used. This means that $\left(\overline{\rho h}\right)_i = \frac{1}{2}\left[(\rho h)_{i+1} + (\rho h)_i\right]$ in Eq. (7.3-13) which becomes,

$$\frac{d}{dt}\left\{\frac{1}{2}\left[(\rho h)_{i+1} + (\rho h)_i\right] \Delta Z\right\} - (\rho h)_{i+1} \dot{Z}_{i+1} + (\rho h)_i \dot{Z}_i = Q_i \Delta Z \tag{7.3-47}$$

Eq. (7.3-47) is written in terms of temperature and both ρ and c_p are assumed to be temperature independent. This results in the following,

$$\frac{d}{dt}\left\{\frac{1}{2}\left(T_{i+1}+T_i\right)\right\}\Delta Z+\frac{1}{2}\left(T_{i+1}+T_i\right)\left(\dot{Z}_{i+1}-\dot{Z}_i\right)-T_{i+1}\dot{Z}_{i+1} \\ +T_i\dot{Z}_i=\frac{\Delta Z}{\rho c_p}Q_i \tag{7.3-48}$$

Since the wall temperature is tracked at the cell center, not the cell edge, the quantity desired is $\overline{T_i}=\frac{1}{2}(T_{i+1}+T_i)$. Thus Eq. (7.3-48) becomes, after simplification,

$$\dot{\overline{T}}=\frac{1}{\rho c_p}Q_i^k+\frac{1}{2}\frac{1}{\Delta Z^{k+1}}\left(T_{i+1}^k-T_i^k\right)\left(\dot{Z}_{i+1}^{k+1}+\dot{Z}_i^{k+1}\right) \tag{7.3-49}$$

The heat source is explicit in time and the $DeltaZ$ and $\dot{Z}$ terms are the end-of-step values from the water side calculation. The difficulty with this equation is calculating the cell-edge values T_{i+1} and T_i since the wall temperatures are tracked at the cell center. Therefore interpolated or extrapolated estimates of the cell-edge values are formed from the cell-center temperatures.

7.3.3.1.7 Calculation of Boiling Crisis Point

The point of boiling crisis, or DNB point, in the boiling zone is computed by tracking the continuously varying intersection of two functions which is a point within the node structure. The first function represents the required heat flux for the boiling crisis to occur and the second is the actual local heat flux at the wall surface.

The DNB heat flux correlation *[7-3]* is as follows,

$$F_D=7.84\times10^8\left[\chi h_{fg}\rho_g/\rho_f\sqrt{\frac{G}{1355}}\right]^{-0.667} \tag{7.3-50}$$

This correlation is evaluated at each cell center in the boiling zone using the local quality. The inlet mass flux G is used instead of the local mass flux to enhance numerical stability although the original correlation used the local flow.

An expression for the wall surface heat flux is obtained as follows. There is a correlation for the heat transfer coefficient at the tube wall surface but the wall surface temperature is unknown, although the mid-wall temperature and the water temperature at saturation are known. Without going into the details of the correlation here (see Section 7.7.2), it is known that the heat flux at the wall surface, F_S, is equal to $a\left(T_S-T_{sat}\right)^2$, where T_S is the wall surface temperature and a is only a function of pressure. The heat flux between the mid-wall and the wall surface, F_M, is $b\left(T_M-T_S\right)$, where b is the inverse of the wall heat resistance and T_M is the mid-wall temperature. When F_S is set equal to F_M, a quadratic in (T_S-T_{sat}) results,

$$a\left(T_S-T_{\text{sat}}\right)^2+b\left(T_S-T_{\text{sat}}\right)-b\left(T_M-T_{\text{sat}}\right)=0 \tag{7.3-51}$$

Thus the wall surface temperature is obtained and then the heat flux at the wall surface, F_S, which is computed at each cell center over the length of the boiling zone. There are thus two functions, F_S and F_D, with values at each cell. In order to obtain the intersection of these two functions and thus the point of boiling crisis, a linear approximation is made to each function proceeding two cells at a time along the length of the steam generator until an intersection of the two lines is reached. The intersection point is tracked exactly and the nucleate boiling and film boiling heat transfer coefficients are prorated in the cell where the intersection occurs. This method gives a smoothly varying, stable calculation of the DNB point.

7.3.3.1.8 Disappearing and Appearing Regions

When the length of a zone is reduced below a certain value, the number of nodes within the zone is reduced from whatever initial number there were in the zone to only one node. However, no matter how long the zone is, no more than the initial number of nodes will be used in the zone. Reducing the number of nodes for small zone lengths greatly enhances numerical stability while reducing computer time and, so long as the criterion for the node reduction is not too large, very little accuracy in the calculation is sacrificed. When the node structure is collapsed to one node, the cell-edge quantities at the inlet and outlet to the zone remain unchanged while the intermediate values are eliminated. The tube wall temperature in the center of the new l-node region is formed as an average of two wall temperatures nearest the center of the old multi-node region. When the zone length increases beyond a certain value and there is only one node in the region then the number of nodes is reset at the original value and the values of parameters at intermediate nodes must be initialized. This is done with simple linear fits (close enough considering the short lengths involved) between the end point values for sodium temperatures, for enthalpies in the subcooled and superheated zones and for mass flows in the boiling and superheated regions. When the boiling zone has only one node and when it is reinitialized at the original number of nodes, the whole region is assumed to be in the nucleate boiling regime.

When a second length threshold is reached as a region's length is reduced, then the region is eliminated entirely. This only applies to the boiling and superheated zones and they must disappear in order. That is if there is a superheated zone, there must be a boiling zone no matter how small. When the superheated zone disappears, then the boiling zone outlet enthalpy (i.e. void fraction or quality) becomes a free variable rather than fixed at h_g as it is when a superheated zone exists. The criterion for elimination of the superheated zone is a length criterion but the criterion for recreating the superheated zone is that the outlet enthalpy of the boiling zone be somewhat higher than h_g. The mass flow at the outlet of the newly created superheated zone is assumed the same as the outlet flow from the boiling zone. The outlet enthalpy is set to the value of the criterion and the boiling zone outlet enthalpy is set to h_g. The length of the new superheated zone is computed according to the following formula,

$$Z_{SH} = Z_{TP} \frac{h_{out} - h_g}{h_{out} - h_f} \tag{7.3-52}$$

where Z_{SH} is the new superheated length, Z_{TP} is the old boiling zone length before being reduced by Z_{SH} and h_{out} is the enthalpy calculated at the outlet of the steam generator. Values for the sodium and tube wall temperatures are interpolated according to the new node and zone structure.

The criterion for elimination of the boiling zone is again a length criterion but the criterion for recreating the boiling zone is that the outlet enthalpy of the subcooled zone be somewhat higher than h_f. The mass flow at the outlet of the newly created boiling zone is assumed the same as the outlet flow from the subcooled zone. The outlet enthalpy is set to the value of the criterion and the subcooled zone outlet enthalpy is set to h_f. The length of the new boiling zone is calculated as follows,

$$Z_{TP} = Z_{SC} \frac{h_{out} - h_f}{h_{out} - h_{in}} \tag{7.3-53}$$

where Z_{TP} is the new boiling length, Z_{SC} is the old subcooled zone length before being reduced by Z_{TP}, h_{out} is the enthalpy calculated at the outlet of the steam generator, and h_{in} is the inlet enthalpy. Values for the sodium and tube wall temperatures are interpolated as for the superheated zone creation above according to the new node and zone structure.

7.3.3.1.9 Miscellaneous Numerical Issues

Nothing has been said so far about the calculation of the time derivative of pressure which appears in the energy Eq. (7.3-15). At each call to the steam generator routine, the balance-of-plant calculation provides the new steam generator pressure at the end of the time step. It would seem natural to form $\dot{P}$ with the current Δt and P^k and P^{k+1}. This, however, can lead to significant numerical instabilities caused by large temporary spikes in P even when the time-averaged value of $\dot{P}$ is well-behaved and much more gradual than would be predicted by using the stepwise values of P. Therefore, a moving average of $\dot{P}$ is computed over the last 2 m timesteps so that,

$$\dot{P} = \frac{\sum_{i=m+1}^{2m} P_i \Delta t_i - \sum_{i=1}^{m} P_i \Delta t_i}{\frac{1}{2}\sum_{i=1}^{2m} \Delta t_i} \tag{7.3-54}$$

There is a time step selector currently in the code which chooses a new timestep size for the next step based on information from the current step. However, this selector is preliminary and merely chooses the new step according to fractional changes in a number of parameters over the step. That is, the new time step is computed as follows,

$$\Delta t' = \text{Min}\left(X \left| \frac{\Delta t Y_i^{k+1}}{Y_i^{k+1} - Y_i^k} \right| \right) \tag{7.3-55}$$

where X is an input fractional change in the quantity Y_i, Δt is the current time step, $\Delta t'$ is the new time step and Min indicates that the minimum over all the Y_i values is computed. The various quantities indicated by Y include the sodium side flow, the zone boundaries, the steam generator pressure, the nodal water mass flows, the void fractions in the boiling zone, the enthalpies in the subcooled and superheated zones, the sodium temperatures and the tube wall temperatures. There is also a minimum value criterion for the new step set by the steam generator. There is in effect a maximum value for the time step which is set for the primary loop calculation. However, there is rarely any need for the steam generator to have a larger time step than the primary loop calculation and, almost always, it is the hydrodynamics on the water side of the steam generator which determines the time step.

There are two artificial limitations that are superimposed on the boiling zone calculation that need to be pointed out. The first is that the void fraction solved for in each successive node as the solution proceeds up the zone must be larger than the previous value at the last node. The code simply requires that the new void fraction be at least 0.001 larger than the last. This may seem like a major limitation in the solution method but, in practice, it is rarely used. When it is used, it has very little consequence for the calculation. The only time this fix is used is when there is an extremely flat void fraction profile (which can be caused by a number of things, very low water flow, for example). In the absence of this fix, there is occasionally a tendency for numerical instabilities to form when there is an extremely flat void profile. The other artificial limitation is that the boiling zone length is not allowed to change more than 1% in a time step. This can have some significant implications for the course of a transient. Without this limitation, there can be a tendency for the boiling zone length to adjust too rapidly to changing conditions, which can cause significant numerical instabilities since a large change in Z_{TP} over a short time produces a large $\dot{Z}_{TP}$ and a large perturbation in the equation set. Since the effect is largely artificial, this limitation within certain bounds is consistent with the physical conditions of the case. However, limiting the change in Z_{TP} means that the void fraction at the top of the zone may not be equal to 1.0 and, if it is too different, then basic assumptions of the model are, of course, being violated. It must be emphasized, however, that this limitation, is not used frequently and, when it is used, the outlet void fraction nearly always remains close to 1.0 while, over a number of time steps, the region length changes to accommodate the changing conditions but avoiding large temporary values of $\dot{Z}_{TP}$ which could drive the calculation unstable.

7.3.3.1.10 Calculation of Pressure Drop for Momentum Equation

Although the balance-of-plant calculation does the actual computation which produces the inlet flow for the steam generator and the pressure boundary conditions at the inlet and outlet plena of the steam generator, the steam generator must provide the pressure drops to the balance-of-plant for this calculation. For the details of the momentum equation solution, it is necessary to refer to the balance-of-plant description (Section 7.2). All that will be done here is to describe the pressure drop calculation itself. For each of the four zones corresponding to each of the four heat transfer regimes i, the following is the pressure drop ΔP across the zone,

$$\Delta P_i = -\overline{G}_i^2 \left(\frac{1}{\rho_t} - \frac{1}{\rho_b} \right) - Z_i \left[\overline{\rho}_i 9.8 + \dot{\overline{G}}_i + \overline{G}_i^2 0.31 F R_i \left(\frac{\overline{G}_i D_H}{\overline{\mu}_i} \right)^{-0.25} \frac{1}{2} \frac{1}{D_H \overline{\rho}_i} R_i \right] \tag{7.3-56}$$

where $\overline{G}_i$ is an average of the mass flow both spatially over the zone length and temporally over the time step; ρ_t is the density at the top of the zone and ρ_b at the bottom; Z_i is the zone length; $\overline{\rho}_i$ the average density over the zone; $\dot{\overline{G}}_i$ is the average mass flow over the zone length at the end of the time step minus the average mass flow over the zone at the beginning of the time step divided by the time step; FR_i is a calibration factor computed in steady state to provide the proper steady state pressure drop; D_H is the water side hydraulic diameter; and $\overline{\mu}_i$ is the average viscosity for the regime. Eq. (7.3-56) is used to compute ΔP_i for the subcooled, the nucleate boiling, the film boiling and superheated zones. The factor R_i is 1.0 for the subcooled and the superheated zones. R_i is computed according to the following formula in the two boiling zones,

$$R_i = 0.95819 - (1167.2 R'_i - 505.)\, R'_i \tag{7.3-57}$$

$$R'_i = \frac{\overline{\chi}_i}{\left(\frac{P}{1.724 \times 10^6} \right)^{2.448} + 16.217} \tag{7.3-58}$$

where P is the steam generator pressure and $\overline{\chi}_i$ is the average quality over each boiling zone. This formula is the Thom correlation *[7-2]* with constants appropriate for the units used in the code.

Once the ΔP_is have been calculated, then the ΔP_is for the boiling zones and the superheated vapor zone are summed and divided by the sum of all four ΔP_is. This gives the current fraction of the total pressure drop across the steam generator that is above the subcooled zone. The balance-of-plant model uses its current pressures at the inlet and the outlet plena of the steam generator to provide the total pressure drop and estimates the pressure at the top of the subcooled zone with the fractional pressure drop referred to above. The pressure at the top of the subcooled zone provides the balance-of-plant momentum equation with a pressure boundary condition and it can merely include the subcooled liquid zone of the steam generator as the last in a series of incompressible liquid segments bounded by plena which stretches from the feed water inlet to the lower end of the compressible zones in the steam generator (i.e. the subcooled/boiling boundary). Thus the inlet flow (constant throughout the subcooled zone) is determined as part of the solution matrix in the total balance-of-plant momentum equation. Of course, the steam generator must provide the balance-of-plant model with information about the subcooled zone. It needs the average density, the length of the zone, the friction normalization factor FR_1, the average viscosity and the hydraulic diameter. It is clear that, in the case when the subcooled zone extends to the top of the steam generator, the whole steam generator becomes merely one incompressible segment in the balance-of-plant matrix from the feedwater inlet to the turbines.

7.3.3.2 Recirculation-Type Steam Generator

As noted before, the modeling of the evaporator in this node of the steam generator model is done with the same coding as is used for the once-through type steam generator. The difference is that, when it is used as an evaporator, the outlet enthalpy will be less than or equal to h_g. Since this is within the envelope of cases for which the once-through modeling was designed, this has already been described above. The modeling of the steam drum is described elsewhere in detail (See Section 7.4).

There remains only the discussion of the modeling of the superheater. As noted above, it is assumed that incompressible flow is adequate to describe the superheater. The mass flow through the superheater is determined by the balance-of-plant momentum equation. The lower enthalpy boundary condition is h_g. There are no region boundaries to calculate since there is only single phase vapor flow in the superheater. Since $\dot{\rho} = 0$ is assumed, the energy Eq. (7.3-15) becomes, for the constant superheater length,

$$\rho_{i+1}\dot{h}_{i+1}\Delta Z = -G\left(h_{i+1} - h_i\right) + Q_i\Delta Z + \dot{P}\Delta Z \tag{7.3-59}$$

The finite difference form of Eq. (7.3-59) is,

$$\rho_{i+1}^k \Delta Z \frac{h_{i+1}^{k+1} - h_{i+1}^k}{\Delta t} = -G^{k+1}\left(h_{i+1}^{k+1} - h_i^{k+1}\right) + Q_i^k\Delta Z + \dot{P}\Delta Z \tag{7.3-60}$$

Eq. (7.3-60) is then solved for h_{i+1}^{k+1} resulting in,

$$h_{i+1}^{k+1} = \frac{\rho_{i+1}^k \frac{\Delta Z}{\Delta t} h_{i+1}^k + G^{k+1}h_i^{k+1} + Q_i^k\Delta Z + \dot{P}\Delta Z}{\rho_{i+1}^k \frac{\Delta Z}{\Delta t} + G^{k+1}} \tag{7.3-61}$$

Eq. (7.3-61) is solved for each successive h_{i+1}^{k+1} up to the top of the superheater. ΔZ is, of course, a constant. The lower boundary conditions are h_g and ρ_g. G^{k+1} is provided by the balance-of-plant momentum equation.

The sodium side temperatures are solved according to the same method as used in the once-through steam generator. All the same assumptions concerning incompressible flow and donor-cell differencing are made. The only difference is that the $\dot{Z}$ terms are eliminated and ΔZ becomes a constant because of the constant superheater length. Thus Eq. (7.3-46) becomes,

$$T_i^{k+1} = \frac{T_{i+1}^{k+1}G^{k+1} + T_i^k \frac{\Delta Z}{\Delta t}\overline{\rho}_i^k + \frac{\Delta Z}{\overline{c}_{p,i}}Q_i^k}{\frac{\Delta Z}{\Delta t}\overline{\rho}_i^k + G^{k+1}} \tag{7.3-62}$$

As before, starting at the top of the steam generator, with the new inlet sodium temperature at the end of the time step, the calculation proceeds downward to the bottom of the mesh.

The same considerations apply to the tube wall temperature calculation in the superheater. The solution method is precisely the same as for the once-through steam generator except that the $\dot{Z}$ terms are now eliminated because of the constant superheater length. Thus Eq. (7.3-49) becomes simply,

$$\dot{\overline{T}}_i = \frac{1}{\rho c_p}Q_i^k \tag{7.3-63}$$

Thus there is no difficulty in calculating the ΔT term as there was in Eq. (7.3-49) because it is eliminated.

7.3.3.3 Steady State Solution

7.3.3.3.1 Once-Through Type Steam Generator

7.3.3.3.1.1 Superheated Vapor Region

For a given reactor power, the product of flow rate and enthalpy change across the steam generator must be the same on both the water and sodium sides. In other words, the following holds,

$$RP = G_w A_w \left(h_{w,out} - h_{w,in}\right) = G_s A_w \bar{c}_{p,s} \left(T_{s,in} - T_{s,out}\right) \tag{7.3-64}$$

where RP is the reactor power (or a fraction thereof for multiple steam generators); G_w and G_s are the water and sodium flow rates; A_w and A_s are the water and sodium flow areas; $h_{w,out}$ and $h_{w,in}$ are the outlet steam and inlet water enthalpies; $T_{s,in}$ and $T_{s,out}$ are the inlet and outlet sodium temperatures; and $\bar{c}_{p,s}$ is the average specific heat for sodium over the length of the steam generator. This relationship must determine the above flow rates, water enthalpies, geometry and sodium temperatures. Constraints on any of these parameters must translate into constraints on the other parameters according to the above relationship. For the present purpose, however, geometry, flow rates and inlet and outlet enthalpies are assumed to have been determined elsewhere.

For the steady state, then, $h_{w,out}$ is presumed. If $h_{w,out} > h_g$, then there is a superheated vapor zone. (Of course, in any true steady-state operation for a once-through steam generator, there will be a superheated vapor zone. However, this "steady-state" calculation described here produces starting conditions for a transient calculation of any nature which may be very different than the normal operational conditions.) For the superheated vapor zone, then, the following relation holds,

$$G_w A_w \left(h_{w,out} - h_g\right) = G_s A_s \bar{c}_{p,s} \left(T_{s,in} - T_{s,hg}\right) \tag{7.3-65}$$

where $T_{s,hg}$ is the sodium temperature at the point of saturated vapor enthalpy on the water side and $\bar{c}_{p,s}$ is the average value of the specific heat over the superheated vapor zone. If $\bar{c}_{p,s}$ is known, then Eq. (7.3-65) can be used to determine $T_{s,hg}$. As a practical matter, $c_{p,s}$ varies only 2-3% and quite smoothly over the length of the steam generator. If Eq. (7.3-65) is solved for $T_{s,hg}$ using $c_{p,s}$ calculated with $T_{s,in}$, then the average of $T_{s,in}$ and $T_{s,hg}$ are used to calculate $c_{p,s}$ and if this process is repeated several times, then there is a negligible error in $\bar{c}_{p,s}$ and thus in $T_{s,hg}$.

In order to fully characterize the superheated zone, either the length of the zone must be specified or some calibrating factor on the water side heat transfer coefficient must be specified. This will become clear as the solution description continues. It is also possible to calibrate the sodium side heat transfer coefficient but it is assumed that the water side coefficient involves much more uncertainty. First, the form of the water side heat transfer coefficient is as follows,

$$H_T = \frac{1}{\frac{1}{H_w CF} + WR + FL} \tag{7.3-66}$$

where H_T is the total water coefficient; H_w is the heat transfer coefficient between the tube wall surface and the bulk fluid; CF is some calibration factor to be determined; WR is the heat resistance of the tube wall; and FL is some additional resistance to take into account fouling at the wall surface. H_w is calculated for each heat transfer regime according to correlations given in the Section 7.5. The H_ws represent values for experimental conditions and therefore may need to be calibrated for full scale cases. WR is calculated from the tube wall geometry and properties. FL is specified by the code user in the input.

Before a full nodewise solution is obtained, a rough guess to initialize the iterative process of the final solution is required. The case of specifying the zone length Z_{SH} and computing the calibration factor is considered first. The following equates the average heat transfer in the zone from the bulk sodium to the tube wall with the portion of reactor power known to be derived from the superheated zone,

$$G_s \frac{1}{Z_{SH}} \bar{c}_{p,s} \left(T_{s,in} - T_{s,hg}\right) = H_s \left(\overline{T}_s - \overline{T}_m\right) \frac{2\pi r_s Z_{SH}}{A_s Z_{SH}} \tag{7.3-67}$$

$T_{s,hg}$ is obtained from Eq. (7.3-64). $\overline{T}_s = \frac{1}{2}(T_{s,in} + T_{s,hg})$. r_s is the outer tube wall radius. $\overline{T}_m$ averaged over the height of the zone. H_s is the total heat transfer on the sodium side including the wall resistance,

$$H_s = \frac{1}{\frac{1}{H_{Na}} + WRNA} \tag{7.3-68}$$

where H_{Na} is the heat transfer coefficient from the tube surface to the bulk sodium and WRNA is the wall resistance on the sodium side of the tube. The correlation for H_{Na} is given in Section 7.7.2. H_{Na} is a function of G_s, geometry and temperature. The temperature used is $\overline{T}_s$. Eq. (7.3-67) is solved for $\overline{T}_m$ which is inserted into the equation analogous to Eq. (7.3-67) on the water side,

$$G_s \frac{1}{Z_{SH}}(h_{w,out} - h_g) = \frac{1}{\frac{1}{H_w CF} + WR + FL}(\overline{T}_m - \overline{T}_w)\frac{2\pi r_w Z_{SH}}{A_w Z_{SH}} \tag{7.3-69}$$

where $\overline{T}_W$ is the temperature derived from $\overline{H}_w = \frac{1}{2}(h_{w,out} + h_g)$, r_w is the inner tube wall radius and H_w is calculated with properties derived from $\overline{h}_w$. Thus Eq. (7.3-69) can be solved for CF, the initial guess for the calibration factor on the superheated zone heat transfer coefficient when the zone length is specified. Alternatively, if the calibration factor CF is specified, and the zone length is unknown, then Eq. (7.3-67) and Eq. (7.3-69) form a set with two unknowns, $\overline{T}_m$ and Z_{SH}, which can be solved for easily. If there is only one node in the superheated zone, then the steady state solution is finished at this point.

In order to solve the nodal equations, a similar method is used for each node as in the 1-node approximation above. First, there is the nodal energy balance for cell i from node i to node $i+1$ (the solution proceeds from the bottom to the top of the mesh),

$$G_w A_w (h_{w,i+1} - h_{w,i}) = G_s A_s \, \overline{c}_{p,s,i}(T_{s,i+1} - T_{s,i}) \tag{7.3-70}$$

where $\overline{c}_{p,s,i}$ is the specific heat corresponding to $\frac{1}{2}(T_{s,i+1} + T_{s,i}) = \overline{T}_{s,i}$ This equation can be solved for $T_{s,i+1}$ which results in

$$T_{s,i+1} = \left[\frac{G_w A_w}{G_s A_s \overline{c}_{p,s,i}}\right] h_{w,i+1} + \left[T_{s,i} - \frac{G_w A_w}{G_s A_s \overline{c}_{p,s,i}} h_{w,i}\right] \tag{7.3-71}$$

which can be written as,

$$T_{s,i+1} = a h_{w,i+1} + b \tag{7.3-72}$$

Next, the sodium side heat transfer is described by the following,

$$G_s \frac{1}{\Delta z}\overline{c}_{p,s,i}(T_{s,i+1} - T_{s,i}) = H_{s,i}\left[\frac{1}{2}(T_{s,i+1} + T_{s,i}) - T_{m,i}\right]\frac{2\pi r_s \Delta Z}{A_s \Delta Z} \tag{7.3-73}$$

where $H_{s,i}$ is calculated according to Eq. (7.3-68) with $\overline{T}_{s,i}$ and $T_{m,i}$ is the cell-center value and ΔZ is of course $\frac{1}{n}Z_{SH}$ with n the number of cells in the zone. Eq. (7.3-73) is solved, for $T_{m,i}$ in the following,

$$T_{m,i} = \left[\frac{1}{2} - \frac{G_s \overline{c}_{p,s,i} A_s}{H_{s,i} 2\pi r_s \Delta Z}\right] T_{s,i+1} + \left[\frac{1}{2} + \frac{G_s \overline{c}_{p,s,i} A_s}{H_{s,i} 2\pi r_s \Delta Z}\right] T_{s,i} \tag{7.3-74}$$

which can be written as,

$$T_{m,i} = c T_{s,i+1} + d \tag{7.3-75}$$

and if Eq. (7.3-72) is substituted into Eq. (7.3-75),

$$T_{m,i} = ach_{w,i+1} + (bc + d) = e h_{w,i+1} + f \tag{7.3-76}$$

Similarly, the water side heat transfer is described by the following,

$$G_w = \frac{1}{\Delta Z}\left(h_{w,i+1} - h_{w,i}\right) = H_{T,i}\left(T_{m,i} - \overline{T}_{w,i}\right)\frac{2\pi r_w \Delta Z}{A_w \Delta Z} \tag{7.3-77}$$

where $H_{T,i}$ is calculated according to Eq. (7.3-66). Next $\overline{T}_{w,i}$ is replaced by $\frac{1}{2}\left[T_{w,i+1} + T_{w,i}\right]$ Then, use is made of the relation,

$$h_{w,i+1} - h_{w,i} = \overline{c}_{p,w,i}\left(T_{w,i+1} - T_{w,i}\right) \tag{7.3-78}$$

Eq. (7.3-78) is solved for $T_{w,i+1}$ and then,

$$\overline{T}_{w,i} = T_{w,i} + \frac{1}{2\overline{c}_{p,w,i}}\left(h_{w,i+1} - h_{w,i}\right) \tag{7.3-79}$$

Eq. (7.3-79) and Eq. (7.3-76) are substituted into Eq. (7.3-77), which results in the following,

$$h_{w,i+1} = \frac{G_w h_{w,i} - H_{T,i}\frac{2\pi r_w \Delta Z}{A_w}\left[f - T_{w,i} + \frac{h_{w,i}}{2\overline{c}_{p,w,i}}\right]}{G_w + \frac{H_{T,i}2\pi r_w \Delta Z}{A_w}\left[\frac{1}{2\overline{c}_{p,w,i}} - e\right]} \tag{7.3-80}$$

When $h_{w,i+1}$ is computed, $T_{m,i}$ and $T_{s,i+1}$ can also be found from Eq. (7.3-76) and Eq. (7.3-72).

Now that the basic method of solution of the nodal equation has been described, several points need to be emphasized. First, in the three equation set, Eq. (7.3-70), Eq. (7.3-73), and Eq. (7.3-77), there are the three unknowns $T_{s,i+1}$, $T_{m,i}$ and $h_{w,i+1}$ which are solved for. However, Eq. (7.3-70) and Eq. (7.3-73) presume that $\overline{c}_{p,s,i}$ is known. Eq. (7.3-73) presumes that $H_{s,i}$ is known and this means that an average sodium temperature for the cell must be known. Eq. (7.3-77) presumes that $H_{T,i}$ and $\overline{c}_{p,w,i}$ are known and various physical properties are required to compute $H_{T,i}$. All of these quantities are functions of $\overline{T}_{s,i}$ and $\overline{T}_{w,i}$ (or $\overline{h}_{w,i}$) which require $T_{s,i+1}$ and $T_{w,i+1}$ (or $h_{w,i+1}$). Therefore, there is an iterative process required to produce better and better values for the various parameters after starting the whole process with some initial estimate of the $T_{s,i}$s and the $h_{w,i}$s. On the very first pass, the nodal values of $T_{s,i}$ and $h_{w,i}$ are simply linear interpolations between the end values but, since there are a number of iterations before the final result is obtained, the effect of this initial assumption is negligible.

Secondly, the main goal of the iterative procedure is to produce either a calibration factor for a specified length or a zone length for a specified calibration factor. The length or calibration factor is simply assumed in the above solution of the nodal equations and when the solution of all the nodal values is complete, then a new length or calibration factor is chosen if the result is not yet acceptable. The nodal solution begins at the point of saturated vapor enthalpy where the $h_{w,i}$ is simply h_g and the $T_{s,i}$ is $T_{s,hg}$, produced from Eq. (7.3-65). It should be mentioned here that Eq. (7.3-65) produces a better and better value of $T_{s,hg}$ on each iteration since the value of $\overline{c}_{p,s}$ used is refined by a recomputation after each iteration by using the new nodal $T_{s,i}$s calculated in the iteration. The nodal solution proceeds upwards from h_g and $T_{s,hg}$ to the top of the steam generator where values of $h_{w,out}$ the $T_{s,in}$ are calculated. At this point, the criterion for convergence needs to be determined. The new values of $h_{w,out}$ and $T_{s,in}$ can both be compared to the externally calculated values. It was decided to use $T_{s,in}$ as the criterion and thus the length or calibration factor is adjusted until the $T_{s,i+1}$ at the top of the steam generator is as close to $T_{s,in}$ as necessary according to an input criterion. There is only one last point to note. This is that for each outer iteration which uses a new length or calibration factor, there are ten inner iterations for each node in the nodal solution which are necessary to converge on values of the heat transfer coefficients and c_ps for an assumed length or calibration factor. If this inner iteration is not done, the outer iteration will not converge.

7.3.3.3.1.2 Subcooled Liquid Region

Because of the difficulty of obtaining a solution for the boiling zone unless the length of the boiling zone is known, the subcooled zone solution, which may not have its length specified, is completed before the boiling zone. Once the lengths for both the superheated vapor zone and the subcooled liquid zone are determined, the length of the boiling zone is, of course, merely the remainder of the steam generator length. There is a boiling zone, however, only when $h_{w,out} > h_f$.

$$G_w A_w \left(h_{w,t} - h_f\right) = G_s A_s \bar{c}_{p,s} (T_{s,t} - T_{s,hf}) \tag{7.3-81}$$

where $h_{w,t}$ is h_g when there is a superheated vapor zone and $h_{w,out}$ when there is not; $T_{s,t}$ is $T_{s,hg}$ when there is a superheated vapor zone and $T_{s,in}$ when there is not; $\bar{c}_{p,s}$ is the average value of the specific heat over the boiling zone. As is similar to Eq. (7.3-65), $\bar{c}_{p,s}$ is first calculated with $T_{s,t}$ and then with an average of $T_{s,t}$ and $T_{s,hf}$ and this process is repeated several times thus producing better and better values of $T_{s,hf}$.

As was the case with the superheated zone, either the length of the subcooled zone or the calibration factor on its heat transfer coefficient must be specified. The total water side heat transfer coefficient is calculated as in Eq. (7.3-66) with a different correlation for H_w and perhaps a different value for FL.

Again, as with the superheated zone, a one-node initial guess for the subcooled zone length Z_{SC} or calibration factor is obtained starting with an equation analogous to Eq. (7.3-67),

$$G_s \frac{1}{Z_{SC}} \bar{c}_{p,s} \left(T_{s,u} - T_{s,out}\right) = H_s \left(\overline{T}_s - \overline{T}_m\right) \frac{2\pi r_s Z_{SC}}{A_s Z_{SC}} \tag{7.3-82}$$

where $T_{s,u}$ is either $T_{s,in}$ when there is no boiling zone or $T_{s,hf}$ which is obtained from Eq. (7.3-81), $\overline{T}_s = \frac{1}{2}\left(T_{s,u} - T_{s,out}\right)$, $\overline{T}_m$ is the midwall tube temperature averaged over the height of the zone and H_s is defined in Eq. (7.3-68). Again $\overline{T}_m$ is obtained from Eq. (7.3-82) and inserted into the analogous equation to Eq. (7.3-69),

$$G_s \frac{1}{Z_{SC}} \left(h_{w,u} - h_{w,in}\right) = \frac{1}{\frac{1}{H_w CF} + WR + FL} \left(\overline{T}_m - \overline{T}_w\right) \frac{2\pi r_w Z_{SC}}{A_w Z_{SC}} \tag{7.3-83}$$

where $h_{w,u}$ is $h_{w,out}$ when there is no boiling zone and h_f when there is a boiling zone and $\overline{T}_w$ is the temperature derived from $\frac{1}{2}\left(h_{w,u} + h_{w,in}\right)$. Eq. (7.3-83) is solved for the initial guess for CF when the zone length is specified. Alternatively, if the calibration factor CF is specified, then the Eq. (7.3-82) and Eq. (7.3-83) form a set with two unknowns, $\overline{T}_m$ and Z_{SC} which are then solved for. If there is only one node in the subcooled zone, the steady state solution is finished here.

In order to solve the nodal equations, the same method is used as was used for the superheated vapor zone except the nodewise solution proceeds from the top of the subcooled zone to the bottom of the steam generator. The nodal energy balance for a cell is exactly the same as Eq. (7.3-70). However, since the direction of solution is from top to bottom, $T_{s,i}$ is solved for,

$$T_{s,i} = \left[\frac{G_w A_w}{G_s A_s \bar{c}_{p,s,i}}\right] h_{w,i} + \left[T_{s,i+1} - \frac{G_w A_w}{G_s A_s \bar{c}_{p,s,i}} h_{w,i+1}\right] \tag{7.3-84}$$

which is written as,

$$T_{s,i} = a h_{w,i} + b \tag{7.3-85}$$

As before, the sodium side heat transfer is described by Eq. (7.3-73). As before, Eq. (7.3-73) is solved for $T_{m,i}$ which results in Eq. (7.3-34). However, since $T_{s,i}$ is the unknown now and not $T_{s,i+1}$, Eq. (7.3-74) is written as,

$$T_{m,i} = c t_{s,i} + d \tag{7.3-86}$$

and Eq. (7.3-85) is substituted into Eq. (7.3-86),

$$T_{m,i} = ach_{w,i} + (bc + d) = eh_{w,i} + f \tag{7.3-87}$$

The water side heat transfer is described by Eq. (7.3-77) just as for the superheated vapor zone, but Eq. (7.3-78) is solved for $T_{w,i}$ and,

$$\overline{T}_{w,i} = T_{w,i+1} - \frac{1}{2\overline{c}_{p,w,i}} \left(h_{w,i+1} - h_{w,i}\right) \tag{7.3-88}$$

Eq. (7.3-88) and Eq. (7.3-87) are substituted into Eq. (7.3-77) which results in the following,

$$h_{w,i} = \frac{G_w h_{w,i+1} - H_{T,i} \frac{2\pi r_w \Delta Z}{A_w} \left[f - T_{w,i+1} + \frac{h_{w,i+1}}{2\overline{c}_{p,w,i}}\right]}{G_w + \frac{H_{T,i} 2\pi r_w \Delta Z}{A_w} \left[e - \frac{1}{2\overline{c}_{p,w,i}}\right]} \tag{7.3-89}$$

As before, when $h_{w,i}$ is computed, $T_{m,i}$ and $T_{s,i}$ can also be found from Eq. (7.3-87) and Eq. (7.3-85). The same general considerations apply to the solution of the subcooled zone equation set as applied to the superheated vapor zone equation set. The criterion for convergence is now that $T_{s,i}$ at the bottom of the mesh be arbitrarily close to $T_{s,out}$. The subcooled zone length or the calibration factor CF is adjusted until this is achieved.

7.3.3.3.1.3 Boiling Zone

If $h_{w,out} > h_f$, then there is a boiling zone. It has already been shown how the sodium temperatures were determined at each end of the zone, $T_{s,in}$ or $T_{s,hg}$ (depending on whether or not there is a superheated zone) and $T_{s,hf}$. The water enthalpies at the ends of the zone are either $h_{w,out}$ or h_g and h_f. Since the lengths of the superheated vapor and subcooled liquid zones have been determined either by input specification or by the steady state calculation before the boiling zone calculation begins, the length of the boiling zone is also determined and it remains, therefore, to adjust the heat transfer in the boiling zone so that with the known zone length the enthalpy condition at the top of the zone is obtained. That is, either $T_{s,in}$ or $T_{s,hg}$ is obtained on the sodium side and $h_{w,out}$ or h_g on the water side, although the actual criterion used, as in the other zones, is the sodium temperature. There is a complicating factor when trying to adjust the heat transfer in the boiling zone which does not exist in the other two zones, however. This is the fact that there are two heat transfer regimes in the boiling zone separated at the DNB point. Both the calibrating factors on both the heat transfer coefficients cannot be adjusted independently or the solution won't converge. Therefore, the calibrating factor is fixed arbitrarily at a constant value in the nucleate boiling zone while the calibrating factor in the film boiling zone is adjusted to obtain the proper outlet conditions. A related problem is to determine where the DNB point is in the zone. This is done by using the method outlined in the transient section as the solution proceeds up the mesh and when the intersection of the two functions is determined by extrapolation of the two functions, the DNB point is predicted there.

The method of solution, then, is to proceed upwards in the mesh node-by-node in the nucleate boiling zone using the known zone length and the assumed heat transfer coefficient calibration factor and extrapolating ahead to determine the DNB point. When the DNB point is found in a particular cell, then the calculation for that cell is repeated in order to use the properly prorated heat transfer coefficient (apportioned between the nucleate and the film boiling coefficients) for that cell. Then the solution switches to the film boiling zone and then proceeds to the top of the boiling zone and compares the sodium temperature obtained at the top to either $T_{s,hg}$ or $T_{s,in}$. Before the calculation proceeds node-by-node through the film boiling zone on the first iteration, however, an initial guess is made of the calibration factor in the zone is made on a one-node basis just as was done with Eq. (7.3-67) and Eq. (7.3-69) in the superheated vapor zone. The only difference is that the sodium and water ΔT and Δh are now appropriate for the boiling zone and $\overline{T}_w$ becomes T_{sat}. The calibration factor applied to the film boiling heat transfer coefficient is adjusted on each iteration until the sodium temperature at the top of the zone satisfies the criterion. If, as the calculation proceeds upwards in the mesh, no DNB point is predicted

before the top of the boiling zone is reached, then it is assumed there is no film boiling zone and the calibration factor in the nucleate boiling zone, otherwise constant, is searched upon until the criterion at the top of the zone is satisfied.

In order to solve the nodal equations, a method similar to that used in the superheated vapor zone is used. First a nodal energy balance equation exactly the same as Eq. (7.3-70) is solved for $T_{s,i+1}$ as in Eq. (7.3-71) and rewritten as in Eq. (7.3-72). Again, $T_{m,i}$ is solved for from the sodium side heat transfer Eq. (7.3-73) resulting in Eq. (7.3-74), rewritten as Eq. (7.3-75) and then Eq. (7.3-76). The water side heat transfer equation is slightly simpler than in the superheated zone because the water temperature is known at each node to be T_{sat}. Eq. (7.3-77) becomes,

$$G_w = \frac{1}{\Delta Z}\left(h_{w,i+1} - h_{w,i}\right) = H_{T,i}\left(T_{m,i} - T_{sat}\right)\frac{2\pi r_w \Delta Z}{A_w \Delta Z} \tag{7.3-90}$$

Eq. (7.3-90) is solved for $h_{w,i+1}$ after $T_{m,i}$ is replaced with $eh_{w,i+1} + f$ from Eq. (7.3-76). The result is the following,

$$h_{w,i+1} = \frac{G_w h_{w,i} + H_{T,i}\frac{2\pi r_w \Delta Z}{A_w}\left(f - T_{sat}\right)}{G_w - h_{w,i}\frac{2\pi r_w \Delta Z}{A_w} e} \tag{7.3-91}$$

As before, when $h_{w,i+1}$ is computed, $T_{m,i}$ is found from Eq. (7.3-76) and $T_{s,i+1}$ from Eq. (7.3-72). The same considerations concerning the calculation of properties and heat transfer coefficients during the iterative process and concerning the inner and outer iterations apply to the boiling zone as apply to the superheated zone.

7.3.3.3.1.4 Friction Factors for Momentum Equation

As mentioned before in the section concerning the calculation of the pressure drops in the steam generator, the factor FR_i for each heat transfer regime in Eq. (7.3-56) needs to be defined for steady state conditions. If the "steady state" calculation does not include all heat transfer regimes, then the FR_i for the missing zones is arbitrarily set to 1.0. First an unnormalized pressure drop ΔP_i^u is calculated for heat transfer regime i,

$$\Delta P_i^u = -G_w^2\left(\frac{1}{\rho_t} - \frac{1}{\rho_b}\right) - Z_i\left[\overline{\rho}_i 9.8 + G_w^2 0.31\left(\frac{G_w D_h}{\overline{\mu}_i}\right)^{-0.25}\frac{1}{2}\frac{1}{D_\mu \overline{\rho}_i}R_i\right] \tag{7.3-92}$$

where all parameters except G_w are defined as in Eq. (7.3-56). Next, a normalized pressure drop ΔP_i^n is calculated,

$$\Delta P_i^n = \Delta P_i^u \frac{\Delta P_{SG}}{\sum_i \Delta P_i^u} \tag{7.3-93}$$

where ΔP_{SG} is the specified pressure drop across the entire steam generator length. The FR_is are calculated as follows,

$$FR_i = \frac{\frac{\Delta P_i^n}{Z_i} + 9.8\overline{\rho}_i + \frac{G_w^2}{Z_i}\left(\frac{1}{\rho_t} - \frac{1}{\rho_b}\right)2D_H\overline{\rho}_i}{0.31 G_w^2\left(\frac{G_w D_H}{\overline{\mu}_i}\right)^{-0.25} R_i} \tag{7.3-94}$$

7.3.3.3.2 Recirculation-Type Steam Generator

Only the superheater steady-state condition needs to be described here since, as was explained above, the evaporator steady-state is described in the once-through steam generator section which can account for the situation with $h_{w,out} \leq h_g$. The steady state solution for the superheater is identical with the steady-state solution for the superheated vapor zone covered in the once-through steam generator section. Obviously the option of the fixed length with the search on the calibration factor is the relevant option for the superheater. The water side outlet enthalpy is specified and the inlet water enthalpy is h_g. The sodium inlet temperature, as before, is externally determined and the sodium temperature at the outlet of the superheater must be determined just as it is in Eq. (7.3-65). The solution proceeds upwards to the top of the superheater and the sodium temperature calculated at the top is compared to $T_{s,in}$ and the calibration factor is adjusted until they are sufficiently close.

7.4 Component Models

Models of power plant heat transfer components, turbine, and relief valve have been implemented in the SASSYS-1 balance-of-plant network model. These models extend the scope of the existing balance-of-plant model in the SASSYS-1 LMFBR systems analysis code to handle nonadiabatic conditions and two-phase conditions along flow paths, and to account for work done across the boundaries of compressible volumes. Simple conservation balances and extensive component data in the form of correlations constitute the basis of the various types of components reported here.

This work is part of a continuing effort in plant network simulation based on general mathematical models. The models described in this section are integrated into the existing solution scheme of the balance-of-plant coding. While the mass and momentum equations remain the same as in Section 7.2 (except for the nozzle, which has different momentum equations), the energy equation now contains a heat source term due to energy transfer across the flow boundary or to work done through a shaft. The heat source term is treated fully explicitly. To handle two-phase conditions, the equation of state is expressed differently in terms of the quality and separate intensive properties of each phase.

Table 7.4.1 lists the various types of component models reported. The models are simple enough to run quickly, yet include sufficient detail of dominant plant component characteristics to provide reasonable results. All heater models have been tested as standalone models except the steam drum, which has been tested as part of a recirculation loop. Also an integrated plant test problem simulating an entire LMR plant was carried out with some of the heater models incorporated, and then the turbine model and the relief valve model were included and tested in similar but separate integrated test problems.

7.4.1 General Assumptions

For the simplicity of the models and for the convenience of modeling the components, certain common assumptions are made in all of the heater models. Additional assumptions necessary for each heater model will be described wherever appropriate. Assumptions for the turbine and relief valve models are given in the sections containing the discussions of these models.

The following assumptions are made in all eight heater models. Flow is incompressible on both shell and tube sides. Any two-phase fluid entering on the shell side instantaneously separates into liquid and vapor, and a new thermal equilibrium is reached immediately. The two-phase interface on the shell side serves as the reference point for the saturation pressure of the heater. The two phases are at a common saturation temperature, and each phase is assumed to be at a uniform enthalpy. The momentum equation governing flow entering and exiting the shell side accounts for elevation pressure differences as gravity heads. Nevertheless, intensive properties such as specific volume and thermophysical properties such as viscosity are taken to be uniform for each phase, neglecting elevation effects, when considering heat transfer coefficients.

Table 7.4.1: Component Models

Heaters:	Deaerator, steam drum, condenser, reheater, flashed heater, drain cooler, desuperheating heater, desuperheater/drain cooler
Rotating Machinery:	Turbine (including nozzle)
Valve:	Relief Valve

Additional assumptions are made for the heaters containing tube bundles. Phase change does not occur within the tube bundle irrespective of the fluid temperature on the shell side. Although two-phase fluid may flow through pipes between two volumes, it is not allowed to be present inside the tube bundle which is the tube side of a heater. In general, the tube bundle is modeled as a single tube. Mass flux and pressure drop in the single tube of the model are the same as in the actual tube bundle, and the mass of the metal tubing is also conserved. These constraints do not allow tube length or surface area to be conserved, and so the tube surface heat transfer area is corrected to simulate the bundle heat transfer area through the use of calibration factors which provide an effective thermal resistance for conduction heat transfer.

Details of each heater will now be described, beginning with the simplest model, the deaerator, and progressing to the more complex models. The turbine and relief valve models then follow to yield a total of ten models discussed in this section.

7.4.2 Deaerator

There are two categories of heaters: open heaters and closed heaters. A deaerator is an open heater which refers to the fact that there is no distinction between tube and shell sides, so that hot fluid and cold fluid entering the heater mix together. Actually, a deaerator consists of a closed volume containing liquid and vapor at saturation conditions and is used to remove dissolved gases from an incoming fluid.

7.4.2.1 Model Description

As shown in Figure 7.4.1, a deaerator is a right circular cylinder standing on end. Due to the existence of the two-phase interface, an appropriate response has to be implemented if the interface rises to the level of a pipe through which fluid enters or exits the heater. In the situation when fluid enters the heater, the calculation will be stopped when the interface rises to the flow opening if the incoming fluid is other than either saturated or subcooled liquid. On the other hand, if fluid flows out of the heater through the pipe, a special treatment is required as follows. Normally, each flow opening is located entirely within either the liquid or the vapor region, with outgoing fluid quality equal to 0 or 1, respectively. However, the two-phase interface may also intersect an opening, causing two-phase fluid to flow out of the heater. Under these circumstances, the model assumes a slip ratio of one for the two-phase fluid and uses area weighting to compute the quality of the exiting fluid.

Consider the shaded area in Figure 7.4.2 as the area occupied by vapor exiting the pipe. Then the ratio of this shaded area to the entire opening is simply the void fraction of the exiting flow, i.e., $\alpha \equiv A_g/A$

Slip ratio is defined as

$$\frac{u_g}{u_f} = \left(\frac{\beta}{1-\beta}\right)\left(\frac{1-\alpha}{\alpha}\right)$$

which reduces to $\alpha = \beta$ when $u_g = u_f$ (slip ratio = 1).

Since β is defined as

$$\beta = \frac{1}{1+\left(\frac{(1-x)v_f}{xv_g}\right)} \tag{7.4-1}$$

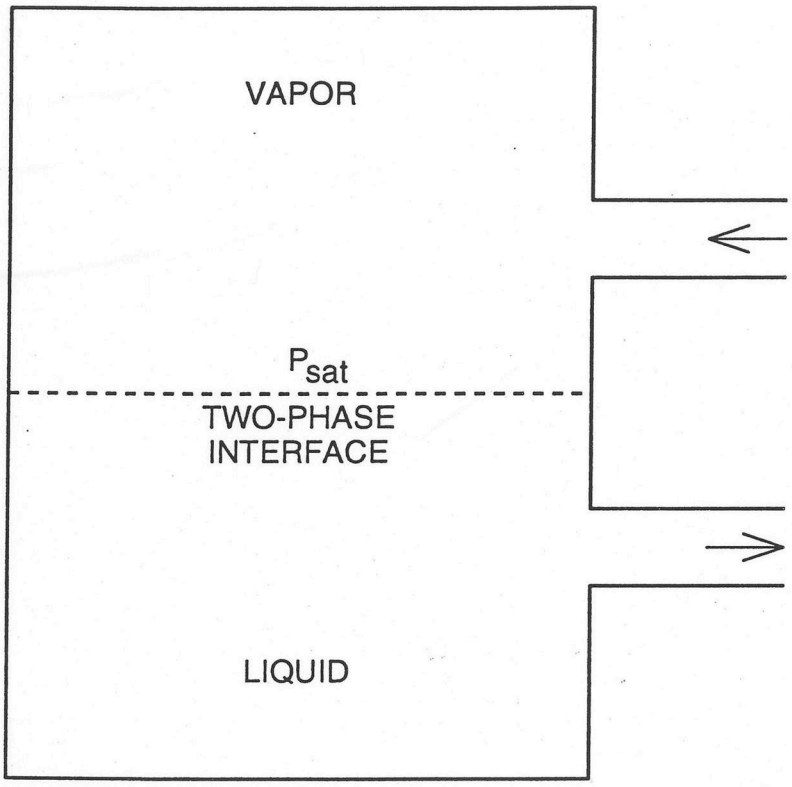


Figure 7.4.1: Deaerator

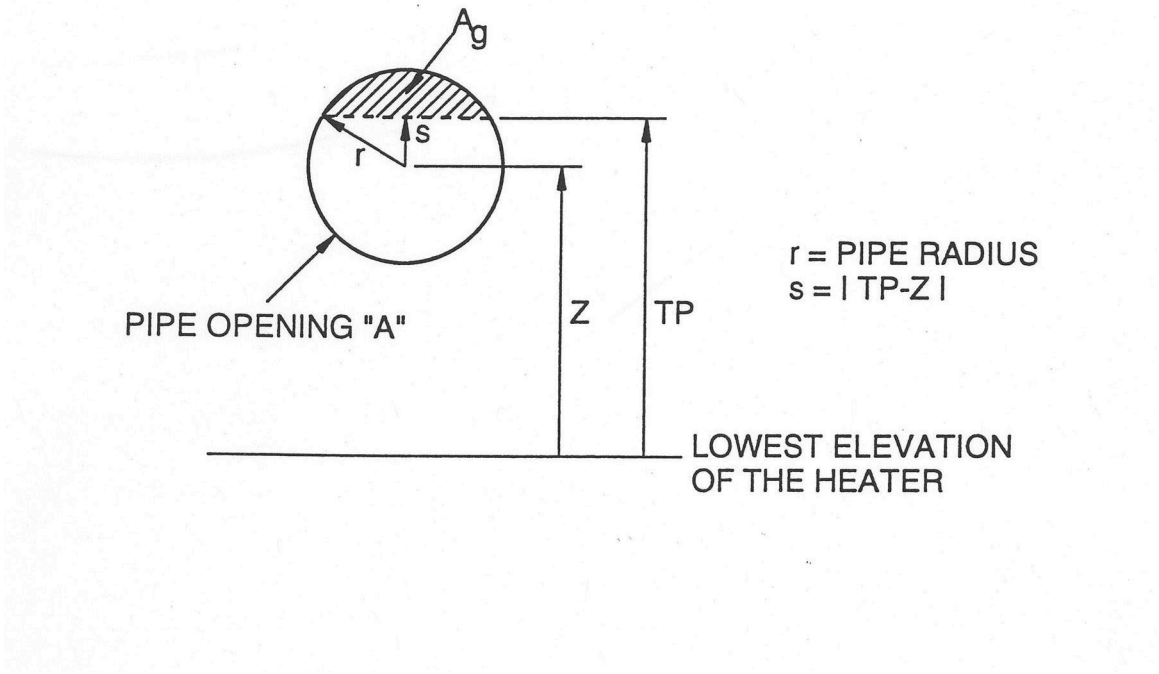


Figure 7.4.2: Area-Weighted Two-Phase Outflow Quality

the quality can be expressed in terms of density and void fraction by rearranging Eq. (7.4-1) and using the fact that $\alpha = \beta$,

$$x = \frac{v_f \alpha}{\left(v_g \left(1-\alpha\right) + v_f \alpha\right)} = \frac{\rho_g \alpha}{\rho} \tag{7.4-2}$$

where

$$\alpha = \frac{r^2 \cos^{-1}\frac{s}{r} - s\left(r^2 - s^2\right)^{1/2}}{\pi r^2} \tag{7.4-3}$$

if the two-phase interface is higher than the center of the pipe but lower than the top of the opening, or

$$\alpha = 1 - \frac{r^2 \cos^{-1}\frac{s}{r} - s\left(r^2 - s^2\right)^{1/2}}{\pi r^2} \tag{7.4-4}$$

if the two-phase interface is lower than the center of the pipe but higher than the bottom of the opening. The above area-weighted quality evaluation method also applies to all of the other heaters for treating two-phase outlet flow.

The two-phase interface of the heater has to be known before deciding the need to evaluate the area-weighted outlet flow quality. To determine the two-phase interface, the shell-side quality, x_s, of the heater has to be found first, which is calculated according to the enthalpy of the heater since the heater is at the saturation pressure, i.e., both the saturation enthalpies of the liquid and the vapor can be obtained from the known saturation pressure. Then, the two-phase interface TP is simply

$$TP = \frac{M\left(1-x_s\right)v_f}{A_s} = \frac{\rho_s A_s H_s \left(1-x_s\right) v_f}{A_s} = \rho_s H_s \left(1-x_s\right) v_f \tag{7.4-5}$$

where M is the heater total mass, A_s is the heater cross section, ρ_s is the heater density, and H_s is the heater height.

The deaerator must cope with small imbalances between incoming and outgoing energies in the steady state. These imbalances are caused by minor inconsistencies between user-specified thermodynamic conditions and the SASSYS-1 correlations for the thermodynamic properties. This problem is solved by the introduction of a pseudo-heat conduction energy transfer term. The temperature gradient between the heater and the ambient conditions serves as the driving force, and the code computes a pseudo-heat transfer coefficient which will give a steady-state energy balance. The coefficient is positive if energy is accumulating and negative if energy is draining. This coefficient is then kept constant throughout the transient. In formula form, the pseudo-coefficient can be expressed as

$$U_{pseu} = \frac{\sum (wh)_{in} - \sum (wh)_{out}}{T_s - T_\infty} \tag{7.4-6}$$

where T_s is the saturation temperature of the heater and T_∞ is the ambient temperature. The other heater models deal with the imbalances this same way.

The thermal-hydraulic effects of noncondensible gas on the model are neglected and the heat conductor term is not applied.

7.4.2.2 Analytical Equations

The applicable mass and energy equations for the deaerator are the same as those used in Section 7.2, so they will not be reiterated here. However, it should be noted that, unlike the volumes described in Section 7.2, there is strict separation of liquid and vapor in the deaerator, and so outgoing flow enthalpies are affected by the elevation of the flow outlet. Calculations of outgoing flow enthalpies must take into account whether the outgoing flow is strictly liquid, strictly vapor, or a two-phase mixture if the outlet intersects the two-phase interface.

The separation of the phases affects the following equation, relating the change in compressible volume pressure to the changes in the flows in each of the segments attached to the volume, which is simply Eq. (7.2-47) in Section 7.2,

$$\begin{aligned} \Delta P_l = -\Delta t \Bigg\{ & Q_l^n + \sum_j w_j^n \left[h_j^n - h_l^n + v_l^n \left(\text{DHDN}\left(l\right) \right) \right] \text{sgn}\left(j,l\right) \\ & + \sum_j \Delta w_j \left[h_j^n - h_l^n + v_l^n \left(\text{DHDN}\left(l\right) \right) \right] \text{sgn}\left(j,l\right) \Bigg\} \; / \; \text{DENOM}\left(l\right) \end{aligned} \tag{7.4-7}$$

The superscript n which denotes the time step, will be omitted for simplicity in the following discussion. Previously, in a volume with perfectly mixed two-phase fluid, the term h_j, which is the enthalpy of segment j attached to volume l, has the same enthalpy as volume l if segment j leads flow out of volume l; h_j is the enthalpy transported along segment j from the volume preceding segment j if fluid flows through segment j into volume l. Now, for the current heater volume, perfect mixing of two-phase fluid no longer exists and immediate separation of liquid from vapor is assumed to take place, so the value of h_j will be different from the mixture enthalpy of volume l due to the existence of the two-phase interface. That is, depending on the relative positions of segment j (pipe) and the two-phase interface, segment j may contain pure vapor, pure liquid, or two-phase fluid with an area-weighted quality determined by the scheme described in Section 7.4.2.1. Therefore, if segment j is an outlet segment, h_j can be $h_{l,g}$, the vapor enthalpy of volume l when the opening of segment j is higher than the two-phase interface; or $h_{l,f}$, the liquid enthalpy of volume l when the opening of segment j is lower than the two-phase interface; or else $h_{l,f} + x(h_{l,g} - h_{l,f})$, where x is the area-weighted quality when the two-phase interface lies within the segment opening. On the other hand, if segment j is an inlet segment, h_j will have to be determined according to which of the three possible situations exists in the volume at the inlet to segment j

rather than in volume l. It is also possible that h_j may just be the transported enthalpy of the preceding volume if the preceding volume is not a heated volume but a volume with fluid of perfect mixing.

For two-phase flow in the pipe, the momentum equation is again basically the same as for single-phase flow except that a two-phase multiplier, which is a function of both quality and pressure, has to be included wherever a two-phase flow is present in the segment (e.g., pipe) to adjust the friction factor for modeling pressure drop. See Section 7.2 for the discussion of this empirically-determined multiplier. One more thing to mention regarding the energy equation is that the source term Q_l in Eq. (7.4-7) is computed explicitly as $(T_s - T_\infty) U_{pseu}$, where U_{pseu} is determined in the steady state by Eq. (7.4-6). T_s is the saturation temperature of the heater in the previous time step and T_∞ is the constant ambient temperature specified by the user in the steady state.

7.4.3 Steam Drum

A steam drum is mainly used to separate two-phase fluid from the recirculation loop and then provide saturated steam to the superheater. Like the deaerator, a steam drum is categorized as an open heater.

7.4.3.1 Model Description

The physical configuration of a steam drum is similar to that of a deaerator except that it is a cylinder now lying on the side, as depicted in Figure 7.4.3. The call for an appropriate response when the two-phase interface falls within the pipe opening as well as the need for coping with small imbalances between incoming and outgoing energies are handled the same way as previously described in the deaerator model.

Predicting the level of the two-phase interface in transients is a difficult problem in a heater of this configuration because the cross-sectional area parallel to the cylinder axis varies in the vertical direction (remember, the cylinder is lying on the side), and so the two-phase interface must be found from a transcendental equation. Since the steam drum is a right circular cylinder, it is obvious from the end view in Figure 7.4.3 that the ratio of A_g to A_g plus A_f is the void fraction α_s, i.e.,

$$\alpha_s = \frac{A_g}{A_g + A_f}$$

where A_g is the projected area occupied by the vapor and A_f is the area by liquid. By taking a similar approach to that carried out in Section 7.4.2.1 and bearing in mind that the cross-sectional area of the entire cylinder is now under consideration, rather than just the opening of a pipe which is attached to the heater as depicted in Figure 7.4.2, equations of the same form as Eq. (7.4-3) and Eq. (7.4-4) are obtained,

$$\alpha_s = \frac{r_s^2 \cos^{-1}\frac{L}{r_s} - L\left(r_s^2 - L^2\right)^{1/2}}{\pi r_s^2}, CV \le TP < CV + r_s \tag{7.4-8}$$

and

$$\alpha_s = 1 - \frac{r_s^2 \cos^{-1}\frac{L}{r_s} - L\left(r_s^2 - L^2\right)^{1/2}}{\pi r_s^2}, CV - r_s \le TP < CV \tag{7.4-9}$$

where $L = |TP - CV|$.

The difference between Eq. (7.4-3) and Eq. (7.4-8) (or Eq. (7.4-4) and Eq. (7.4-9)) is that s on the right-hand side of Eq. (7.4-4) is a known quantity and α is to be found, which in turn yields the pseudo-quality x for the outgoing flow, whereas in the current situation, L in Eq. (7.4-8) is an unknown and α_s is already known from the corresponding quality x_s by the relation $\alpha_s = x_s v_g / v$. The quality x_s is simply calculated from the updated heater pressure and enthalpy. Thus, Eq. (7.4-8) or Eq. (7.4-9) becomes a transcendental equation to be solved for the two-phase interface, TP.

A noniterative scheme has been developed to solve this equation. See Section 7.7.3 for a description of this scheme.

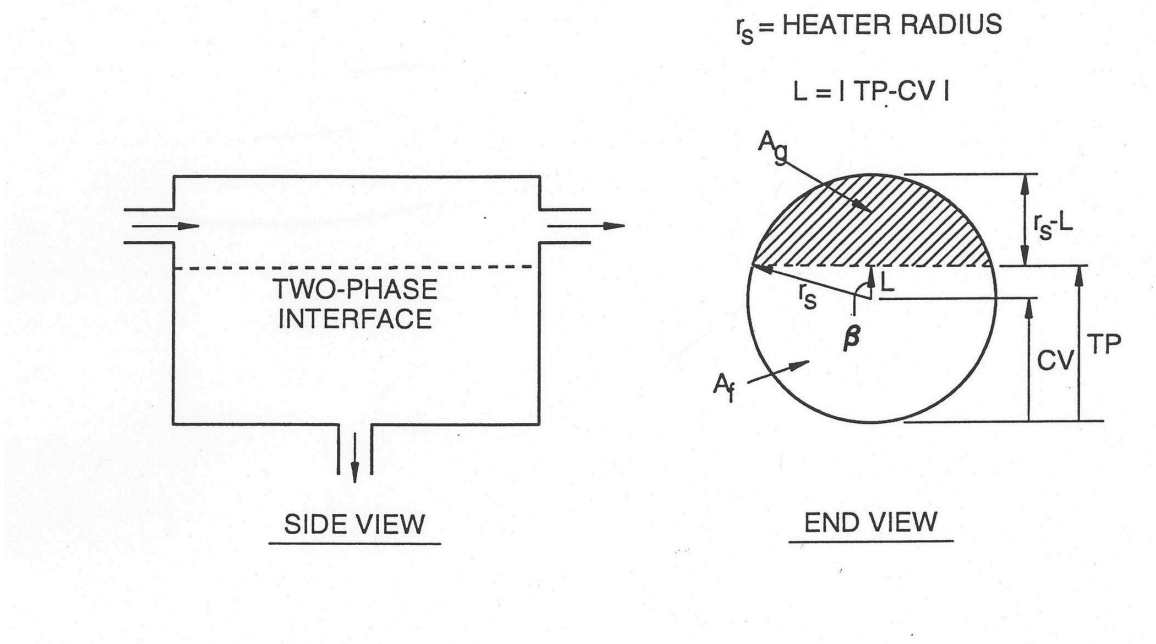


Figure 7.4.3: Steam Drum

7.4.3.2 Analytical Equations

As is obvious from the similarity in the configurations of the deaerator and the steam drum, what is described in Section 7.4.2.2 about the analytical equations for the deaerator is also applicable to the steam drum model.

7.4.4 Condenser

Beginning with this section, a total of six closed heaters including condenser, reheater, flashed heater, drain cooler, desuperheating heater, and desuperheater/drain cooler will be described. The term "closed heater" indicates that hot and cold fluids are separated between a shell side and a tube side. Heat transfer occurs across the tube without contact between hot and cold fluids. Closed heaters consist of a closed volume, or shell side, and a tube bundle, or tube side. Flow is carried into and out of the tube bundle by pipes which lie outside the heater boundary. The condenser is the simplest one among these closed heaters, and it is used to convert steam from the turbine to liquid.

7.4.4.1 Model Description

As diagramed in Figure 7.4.4, a condenser is a box with a tube bundle passing horizontally through it; this is essentially a deaerator with a tube bundle running through the vapor region. The tube bundle may have bends in it. The fluid on the tube side is assumed to be single phase. The tube bundle is assumed to be contained entirely in the vapor region, so a condensation heat transfer coefficient is used on the shell side under the normal conditions when the temperature on the tube side, in which river water or cooling fluid flows, is lower than the shell-side vapor temperature. The model also includes the contingency to use a shell-side heat transfer coefficient computed from the Dittus-Boelter equation (Ref. *7-4*),

$$h = 0.023 \frac{k}{D_h} Re^{0.8} Pr^{0.4} \tag{7.4-10}$$

in case the tube-side temperature is higher than that on the shell side, as in an accident or any unfavorable transients. In either case, the heat transfer coefficient must be adjusted in the steady state so that the tube-side temperature distribution is consistent with the temperatures in the remainder of the plant, i.e., the temperatures at the tube inlet and outlet coincide with the user-specified temperatures.

The heat transfer coefficient on the tube side is calculated using the Dittus-Boelter equation. Heat transfer between the tube side and the shell side is assumed to take place through the mechanism of radial conduction only; axial conduction is neglected. This radial conduction assumption for the tube bundle is made for all of the closed heater models.

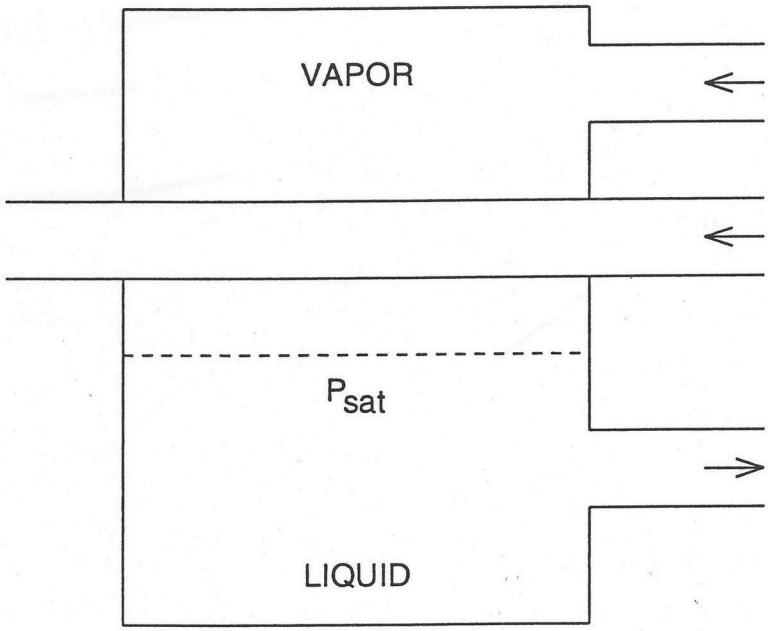


Figure 7.4.4: Condenser

7.4.4.2 Analytical Equations

The mass and energy equations for the shell side of the condenser are the same as those used in the open heaters, and these equations are also valid for the shell side in all the closed heaters. Also, the same momentum equation used in Section 7.2 is again applicable to the tube side (the heated section of the flow segment) of the condenser and other closed heaters. Naturally, the two-phase friction multipliers should be included whenever an inlet or outlet pipe contains other than single-phase fluid. In addition, a simplified energy equation based on the first law of thermodynamics is needed for the heated tube element, i.e., the tube side fluid and the metal tube itself.

For the tube side fluid, the energy equation can be expressed as

$$m_t \frac{dh_t}{dt} = Q_t + w(h_{in} - h_{out}) \tag{7.4-11}$$

if the kinetic energy and potential energy are negligible.

For the metal tube, the energy balance has the form,

$$m_m c_m \frac{dT_m}{dt} = Q_m \tag{7.4-12}$$

7.4.4.3 Discretized Equations

The tube side is discretized as shown in Figure 7.4.5, and the temperature distribution on the tube side is determined using the differenced form of the first law of thermodynamics,

$$m_{it} \frac{\Delta h_{it}}{\Delta t} = Q_{it} + w\left(h_{i-1,t} - h_{it}\right) \tag{7.4-13}$$

where

$$Q_{it} = \left(T_{im} - T_{it}\right) \; / \; \left(\frac{1}{2\pi \Delta z r h_t} + \frac{\ln\left(1 + \frac{\delta}{2r}\right)}{2\pi \Delta z k_m} \right) \equiv \frac{\left(T_{im} - T_{it}\right)}{R_t} \tag{7.4-14}$$

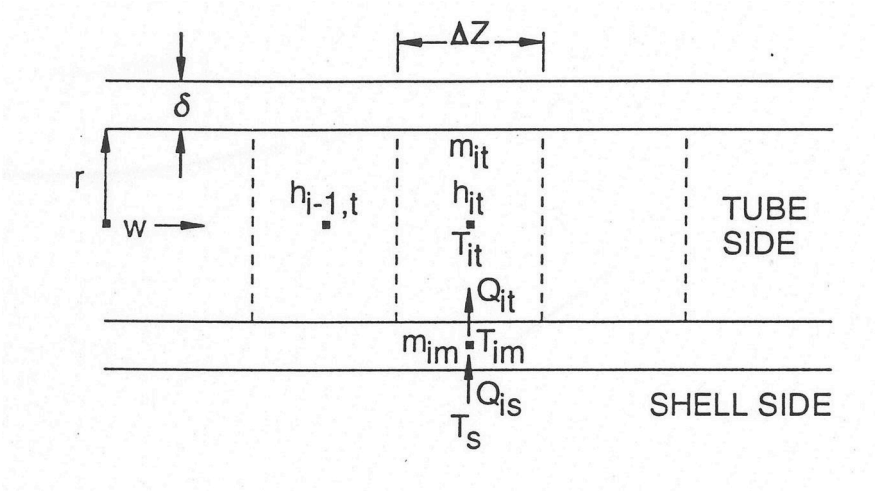


Figure 7.4.5: Tube-side Nodalization

The metal tube temperatures T_{im} are computed from a similar set of equations,

$$m_{im} C_m \frac{\Delta T_{im}}{\Delta t} = Q_{is} - Q_{it} \tag{7.4-15}$$

and

$$Q_{is} = \left(T_s - T_{im}\right) / \left(\frac{\left(\ln \frac{r+\delta}{r+\delta/2}\right)}{2\pi \Delta z k_m} + \frac{1}{2\pi \Delta z \left(r + \delta\right) h_{is}} \right) \equiv \frac{\left(T_s - T_{im}\right)}{R_{is}} \tag{7.4-16}$$

Note that T_s, the shell-side temperature, is considered to be uniform throughout the entire heater volume, whether in the steam region or in the liquid region, and that the specific heat and the thermal conductivity of the metal tube are assumed to be constant within the temperature range under consideration. The thermal resistance on the tube side, R_t, is the same at each node along the tube side but on the shell side, R_{is}, may vary

node to node along the shell side if the tube is partially submerged in the liquid region. When the shell side is hotter than the tube side, it is assumed that no local boiling occurs on the tube side and that the condensate does not form a wetted perimeter along the tube outer periphery.

The updated fluid enthalpy at each node is calculated explicitly using quantities at the previous time step, so from Eq. (7.4-13) it can be expressed

$$h_{it}^{n+1} = h_{it}^{n} + \frac{\Delta t^n}{m_{it}^n}\left(Q_{it}^n + w^n\left(h_{i-1,t}^n - h_{it}^n\right)\right) \tag{7.4-17}$$

where

$$Q_{it}^n = \frac{(T_{im}^n - T_{it}^n)}{R_t^n} \tag{7.4-18}$$

Similarly, Eq. (7.4-15) can be rewritten in explicit form for updating the metal tube temperatures at each node,

$$T_{im}^{n+1} = T_{im}^n + \frac{\Delta t^n}{m_{im}C_m}\left(Q_{is}^n - Q_{it}^n\right) \tag{7.4-19}$$

where

$$Q_{is}^n = \frac{(T_s^n - T_{im}^n)}{R_{is}^n} \tag{7.4-20}$$

The superscript n in R_t^n denotes that the tube side heat transfer coefficient h_t is a function of time, i.e., h_t^n, whereas the metal thermal conductivity k_m is assumed to be constant throughout the transient. This is also true for h_{is} and k_m in R_{is}.

The tube-side fluid temperature T_{it}^n in Eq. (7.4-18) is readily computed according to the equation of state from the fluid enthalpy and pressure at each node, and T_s^n is just the saturation temperature of the shell side fluid in the heater.

All of the terms on the right-hand side of Eq. (7.4-17) are known from the previous time step, so h_{it}^{n+1} can be updated, which will then yield T_{it}^{n+1} for use in Eq. (7.4-18) in the next time step. T_{im}^{n+1} on the other hand is determined independently from Eq. (7.4-19), with Q_{is}^n and Q_{it}^n given respectively by Eq. (7.4-18) and Eq. (7.4-20). The link between the condenser model and the rest of the balance-of-plant models is made through the heat source terms, i.e., Q_l^n in Eq. (7.4-7) and Q_l^n in Eq. (7.4-20). The summation of Q_{is}^n at each node along the shell side is in fact the negative of the heat source term Q_l^n for the heater in Eq. (7.4-7). The sign change reflects the fact that Q_{is} should be a heat sink term for the heater when the shell side temperature is higher than the tube side temperature.

7.4.5 Reheater

A reheater is used to improve turbine performance by reheating the moist steam to the superheated phase as it passes between stages of the turbine.

7.4.5.1 Model Description

The basic features of a reheater are depicted in Figure 7.4.6, for illustrative purposes it is a right circular cylinder standing on end. The shell side is assumed to be at a lower temperature than the tube side. In this type of heater, the shell side is all vapor, and the reheater appears to be little more than the vapor section of a condenser with a vertically bent tube bundle. However, there is one important difference: the fluid on the tube side changes phase from steam to two-phase as it passes through the reheater. Modeling this phase transition requires use of the energy equation; however, the energy equation in the balance-of-plant formulation is solved at flow junctions

(such as the shell side of a heater), not along flow paths (such as the tube side). Therefore, the reheater is modeled in the configuration shown in Figure 7.4.7. The shell side of Figure 7.4.6 becomes the tube side in Figure 7.4.7, with steam which is being heated flowing within the tube, and the tube side of Figure 7.4.6 becomes the shell side in Figure 7.4.7. The shell side of Figure 7.4.7 then easily models the phase transition which occurs in the tube side of the reheater.

The reconfiguration of the reheater to Figure 7.4.7 is done so as to conserve volume on both sides of the heater. The height, H, of the cylinder in Figure 7.4.7 is now taken as the difference between the elevations of the highest tube and the lowest tube in the tube bundle of the original configuration in Figure 7.4.6. Therefore the cross sectional area A_s on the shell side in Figure 7.4.7 is determined as

$$A_s = \frac{V_t}{H}$$

where V_t is the internal volume of all tubes in the tube bundle in Figure 7.4.6. Furthermore, the mass of the metal tube, m_{im}, in each node in Figure 7.4.7 is obtained by dividing the total metal mass of all original tubes, M_m, by H and then multiplying by the node length Δz,

$$m_{\text{im}} = \frac{M_m}{H}\Delta z \tag{7.4-21}$$

Similarly, the internal tube surface area in each node is computed based on the total internal heat transfer surface of all tubes, A_t, in Figure 7.4.6 to be $(A_t/H)\,\Delta z$. Both the inside radius and the outside radius of the tube in Figure 7.4.7 are also needed for use in equations like Eq. (7.4-14) and Eq. (7.4-16) in order to compute heat transfer across the tube boundaries. The inside radius is readily calculated using the internal tube surface area in each node just described above. From the relationship, $2\pi r\Delta z = (A_t/H)\,\Delta z$, the inside radius r_i is computed as

$$r_i = \frac{A_t}{2\pi H}$$

and the outside radius can then be deduced by making use of Eq. (7.4-21) to be

$$r_o = \left(r_i^2 + \frac{M_m}{H\rho\pi}\right)^{1/2}$$

where ρ is the metal density which is assumed to remain the same throughout the reconfiguration.

The model of Figure 7.4.7 is very similar to a condenser except that the tube side is vertical instead of horizontal. This means that the tube passes through both liquid and vapor. Since the tube side vapor is heated by the shell side fluid, the heat transfer coefficient on the shell side h_{is} is computed from the Dittus-Boelter correlation in the liquid region and a condensation coefficient in the vapor region. Usually, the two-phase interface will fall within one of the nodes which discretizes the tube side; in this case, the heat transfer coefficient within the node is an area-weighted combination of the condensation coefficient and the Dittus-Boelter coefficient.

7.4.5.2 Analytical Equations and Discretized Equations

The equations discussed in Section 7.4.4.2 and Section 7.4.4.3 for the condenser model also apply to the reheater model and to the remaining closed heater models. The only difference between the energy equations for the condenser and reheater models is in the calculation of the shell-side heat transfer coefficient, as described in Section 7.4.5.1.

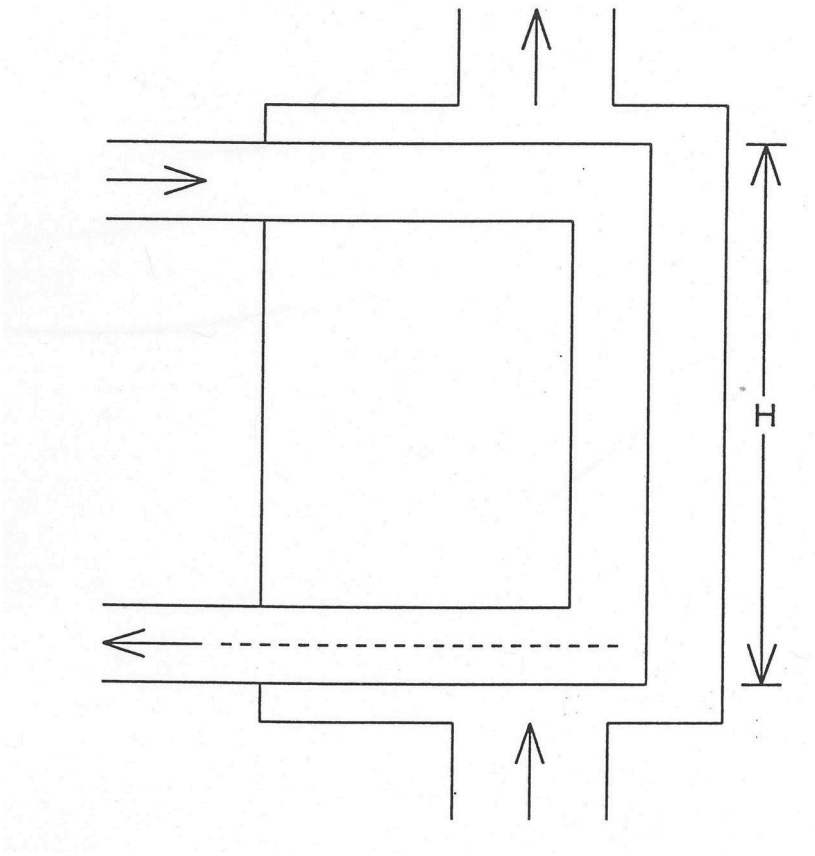


Figure 7.4.6: Reheater

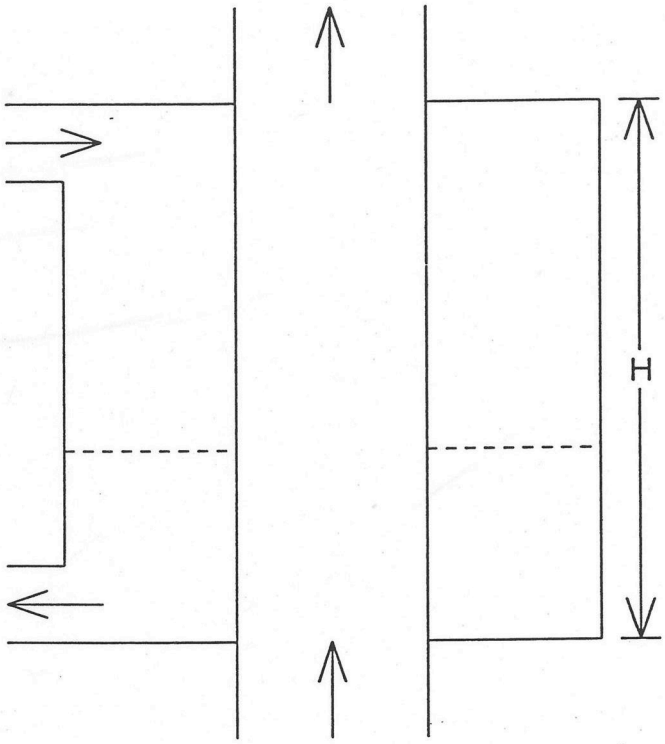


Figure 7.4.7: Reconfiguration of the Reheater

7.4.6 Flashed Heater

A flashed heater is needed when liquid upstream of the heater shell side is above the shell side saturation point. The flashed heater allows the liquid to flash safely upon entering the heater.

7.4.6.1 Model Description

The geometry of the flashed heater is given in Figure 7.4.8. This is a right circular cylinder lying on the side, with a U-shaped tube bundle that is partially submerged in liquid and partially surrounded by vapor. The bundle enters and leaves the shell side through one end of the cylinder, with the entrance and exit at two different elevations. The tube-side fluid is assumed single phase, and fluid can enter at either the lower or the upper elevation. These assumptions are also made in the drain cooler, desuperheating heater, and desuperheater/drain cooler models to be described in subsequent sections. Therefore, there are two bends in the tube, and the tube is considered to consist of three sections: two horizontal ones of equal length and a vertical one of length equal to the distance between the elevations of the tube entrance and exit.

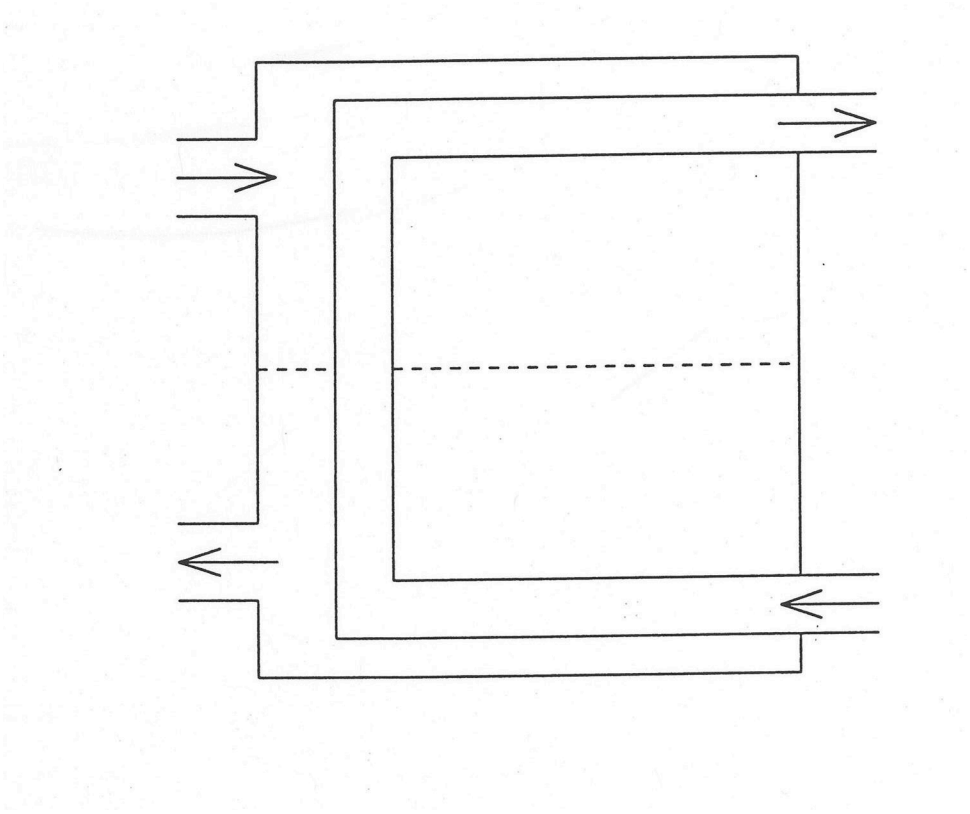


Figure 7.4.8: Flashed Heater

7.4.6.2 Analytical Equations and Discretized Equations

As discussed in Section 7.4.5.2 for the reheater model, the equations of Section 7.4.4.2 and Section 7.4.4.3 also apply to the flashed heater model. However, some details specific to the flashed heater need further explanation. The determination of the heat transfer coefficient on the shell side becomes more complicated due to the bending of the tube. The two-phase interface will not only fall within one of the nodes as before in the reheater, but also might fall within many of the nodes when it reaches either of the two horizontal sections of the tube.

In the latter situation, the scheme of area-weighted heat transfer coefficients is again used. In either of the situations, provision is made to handle the case when the tube side temperature is higher than the shell side temperature, an additional possibility likely to occur during transients. If the tube side is cooler than the shell side, the heat transfer coefficients along the tube surface are computed as for the reheater. If the tube side is hotter than the shell side, the coefficient within the vapor region is calculated from the Dittus-Boelter correlation, and in the liquid region, a boiling heat transfer coefficient from the nucleate boiling regimes (Ref. *7-9*)

$$h = \left(e^{P/(8.7\times10^6)}/22.65\right) q^{0.5} \tag{7.4-22}$$

is used. The tube side heat transfer coefficient is always calculated from the Dittus-Boelter correlation no matter if the tube side is cooler or hotter than the shell side, i.e., assuming no phase change on the tube side in any case.

Predicting the level of the two-phase interface is a complicated problem in a heater of this configuration for two reasons. First, as in the steam drum, the cross-sectional area parallel to the cylinder axis varies in the vertical direction, and so the two-phase interface must be found from a transcendental equation. See Section 7.7.3 for the noniterative scheme to solve this equation. Second, the volume taken up by the tube bundle must be considered when determining the two-phase interface. In order to simplify the calculations, the void fraction is taken to be the ratio of the vapor volume divided by the vapor volume plus the liquid volume (rather than dividing by the total volume, which is the sum of the volumes of vapor, liquid, and tube bundle) when the interface falls between the bundle inlet and outlet, i.e.,

$$\alpha = \frac{x\nu_g}{x\nu_g + (1-x)\,\nu_f} \tag{7.4-23}$$

where x is known once the new volume pressure and enthalpy are updated. Eq. (7.4-8) (or Eq. (7.4-9)) can then be combined with Eq. (7.4-23) to solve for the two-phase interface.

7.4.7 Drain Cooler

When the shell side outlet liquid from a heater must be sufficiently subcooled so as to remain liquid at the lower pressure of a downstream component, a drain cooler is needed.

7.4.7.1 Model Description

As shown in Figure 7.4.9, the drain cooler configuration is identical to that of the flashed heater with the addition of a drain built into a lower corner of the cylinder. The top of the drain extends horizontally across the cylinder perpendicular to the cylinder axis (see end view in the same figure). The drain is separated from the remainder of the shell side except for a flow inlet which brings saturated liquid from the heater into the drain. This flow inlet is assumed to be always submerged in the saturated liquid, and calculation will be stopped whenever the two-phase interface drops below the inlet and uncovers it, since this might cause the subcooled fluid to flow out of the drain and into the shell side of the heater. There is also a flow outlet which carries liquid out of the drain and away from the heater. Liquid within the drain is always assumed to be subcooled at the saturation pressure of the shell side of the heater. Inlet and outlet flows are assumed to be nearly equal during a transient, for no phase-change is allowed in the drain and thus flow may be assumed incompressible. One end of the tube bundle passes through the drain, as seen in Figure 7.4.9. The tube-side fluid can enter at either the lower or the upper elevation.

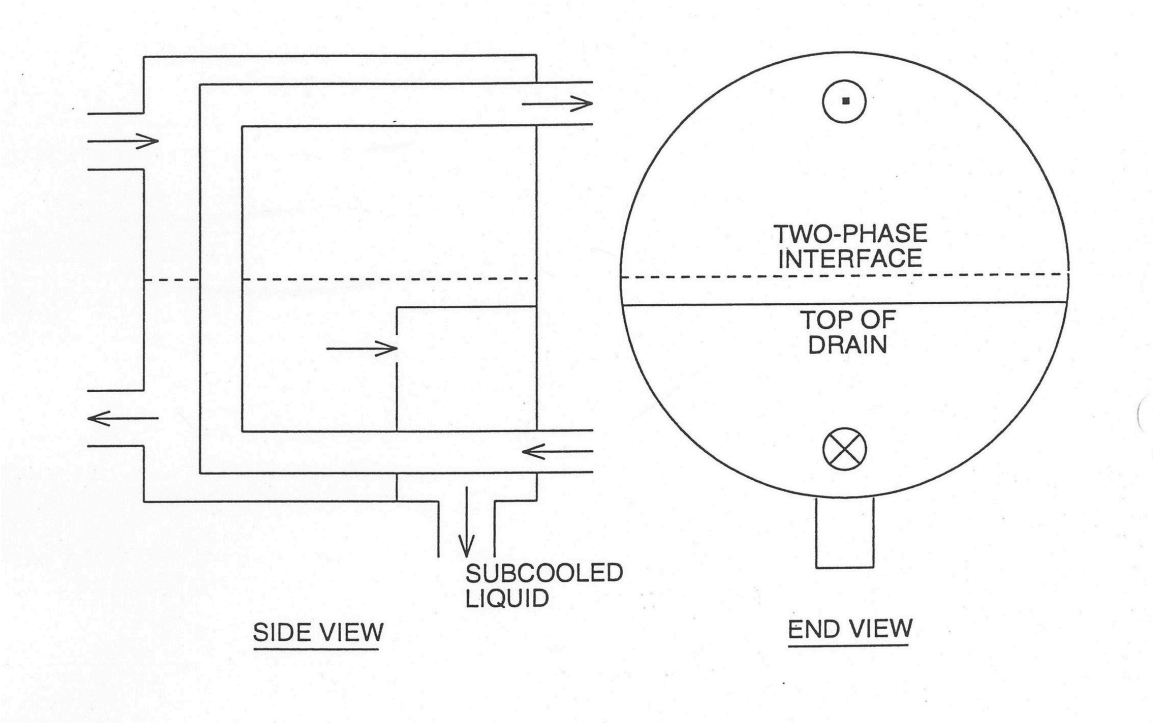


Figure 7.4.9: Drain Cooler

7.4.7.2 Analytical Equations and Discretized Equations

For the part of the drain cooler outside the drain, the discussion in Section 7.4.6.2 about the flashed heater is applicable to the drain cooler model. Nevertheless, there are two further points to be noted. First, although the fluid flowing out of the drain through the pipe which is attached to the drain has the same enthalpy as that of the subcooled liquid in the drain, the energy being lost from the shell side of the heater is actually the saturation enthalpy of the shell side liquid. Therefore, the term h_j^n in Eq. (7.4-7) has to be equal to the saturated liquid enthalpy. When j corresponds to the outlet flow segment from the drain in order for this equation to compute the shell-side pressure change correctly. On the other hand, the fluid enthalpy being transported from the drain to the next component must be set to the subcooled liquid enthalpy of the drain. Second, the heat source term Q_l in Eq. (7.4-7) will now have contributions only from the heat transfer through the boundaries of the tube bundle outside the drain. The energy heat transfer occurring across the tube boundaries within the drain will be considered as Q_D of the drain to be used in Eq. (7.4-24) below.

A one-node energy equation,

$$m_D \frac{dh_D}{dt} = Q_D + W_D(h_{in} - h_{out}) \tag{7.4-24}$$

is currently used on the shell side along the tube within the drain. Plans are to add a multi-node energy equation on the shell side to improve the treatment of the energy balance within the drain, which is especially important in the counterflow situation. On the tube side, the multi-node treatment is retained.

In discretized form, Eq. (7.4-24) is rewritten as,

$$h_D^{n+1} = h_D^n + \frac{\Delta t^n}{m_D^n}\left(Q_D^n + w_D^n\left(h_{in}^n - h_{out}^n\right)\right) \tag{7.4-25}$$

where h_{in}^n is the saturated liquid enthalpy and h_{out}^n in the current one-node treatment, is actually h_D^n. Q_D^n is the

summation of the contributions from all the nodes within the drain, i.e.,

$$Q_D^n = \sum_i Q_{iD}^n \tag{7.4-26}$$

with

$$Q_{iD}^n = \frac{T_{iD}^n - T_{im}^n}{R_{iD}^n} \tag{7.4-27}$$

where

$$R_{iD}^n = \frac{\ln\left(\frac{r+\delta}{r+\delta/2}\right)}{2\pi\Delta z k_m} + \frac{1}{2\pi\Delta z\,(r+\delta)\,h_{iD}^n} \tag{7.4-28}$$

Note that R_{iD}^n is simply R_D^n and T_{iD}^n merely T_D^n in the current one-node treatment on the shell side within the drain, which means h_{iD}^n is equal to h_D^n. Q_D^n is calculated explicitly.

For the tube side within the drain, Eq. (7.3-17) and Eq. (7.4-18) are used, and Eq. (7.4-19) is applied to the metal tube within the drain, with Q_{is}^n in Eq. (7.4-15) replaced by Q_{iD}^n from Eq. (7.4-27).

Within the drain, a heat transfer coefficient h_D^n is computed from the Dittus-Boelter equation on the shell side of the tube surface, regardless of whether the tube side is hotter or colder than the shell side.

Adjustments are made separately to the drain and to the remainder of the shell side in order to achieve a steady-state energy balance which is consistent with conditions in the remainder of the plant. First, the code computes a calibration factor to adjust the tube surface heat transfer area within the drain. Then, energy is balanced in the remainder of the heater by computing a separate factor plus a pseudo-heat transfer coefficient as described previously.

7.4.8 Desuperheating Heater

A desuperheating heater is needed when the steam entering the heater shell side is highly superheated. The entering steam is initially contained in a desuperheating region, where it transfers heat to the tube side fluid rather than dissipating heat to the shell side saturated steam. Steam moving from the desuperheating region to the main section of the shell side is near saturation. The desuperheating region also protects the remainder of the heater from being damaged by highly superheated steam.

7.4.8.1 Model Description

The desuperheating heater is diagrammed in Figure 7.4.10; it can be summed up as a drain cooler turned upside down. Instead of a drain at the bottom, it has a desuperheating region built into an upper corner. The desuperheating region is filled with superheated vapor at the saturation pressure of the heater two-phase interface. Normally, the tube side is filled with a single-phase fluid which is cooler than the vapor in the desuperheating region; the model also allows the tube side temperature to be higher than the shell side temperature, within or outside the desuperheating region. The desuperheating heater model does not handle the situation when the two-phase interface reaches the outlet of the desuperheating region, which might cause the desuperheating region to be partially filled with liquid. If the two-phase interface reaches this point, the calculation is stopped. All other aspects of the drain cooler model, such as inlet and outlet flows of the drain are nearly equal and the tube-side fluid can enter at either of the tube ends, apply to the desuperheating heater.

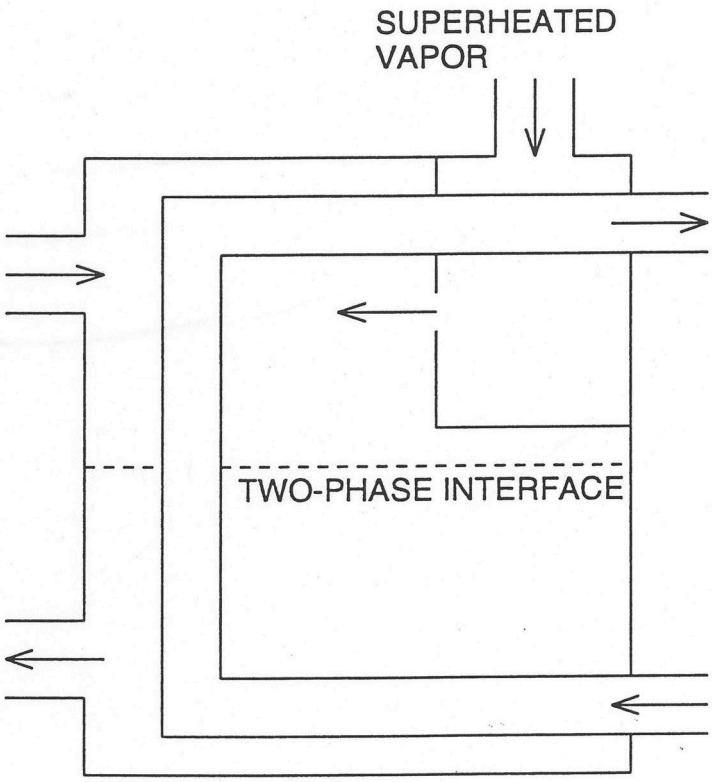


Figure 7.4.10: Desuperheating Heater

7.4.8.2 Analytical Equations and Discretized Equations

All the aspects described in Section 7.4.7 for the drain cooler model are applicable to the desuperheating heater and, wherever necessary, the subscript D for the drain should be replaced by DS for the desuperheating region, such as in Eq. (7.4-24) to Eq. (7.4-28). However, it should be noted that the term h_j^n in Eq. (7.4-7) now is the superheated steam enthalpy of the desuperheating region when j corresponds to the inlet flow segment into the desuperheating region.

7.4.9 Desuperheater/Drain Cooler

7.4.9.1 Model Description

This heater is pictured in Figure 7.4.11 and is a combination of the drain cooler and the desuperheating heater. All details discussed in Section 7.4.7.1 and Section 7.4.8.1 for these two models apply also to the desuperheater/drain cooler.

7.4.9.2 Analytical Equations and Discretized Equations

All aspects described in Section 7.4.7.2 and Section 7.4.8.2 for the drain cooler and the desuperheating heater are applicable to the desuperheater/drain cooler. Three additional points to note are that 1) the mass flow rate entering the desuperheating region can differ from that entering the drain, 2) separate calibration factors are used in the desuperheating section and in the drain for adjusting the tube surface heat transfer area to conserve energy, and 3) the heat source term Q_l in Eq. (7.4-7) will now have contributions only from the energy transferred across the boundaries of the tube bundle outside the desuperheating region and the drain.

7.4.10 Turbine

A turbine is a device in which energy is removed from the fluid as a result of work performed by the flow; it actually is composed of many stages driving one rotor which extracts work from the flow. The stages are connected by nozzles which permit both non-choked and choked flow. Compressible flow is now very important in describing the flow behavior in the nozzles.

7.4.10.1 Model Description

The basic features of a turbine are depicted in Figure 7.4.12. This schematic shows two of the stages connected by nozzles driving one common rotor which in turn drives a generator. Also shown in the figure are the extraction steam ports, in which the steam is treated as incompressible flow. A series of volumes is used to model the various stages in the turbine, and each individual stage is represented by a compressible volume. There is no limit on the number of turbine stages so long as it does not exceed the maximum number of compressible volumes allowed in the code. Nozzles are modeled by special segments using a different form of the momentum equation from that discussed in Section 7.2 in order to account for the characteristics of the compressible flow. Separate expressions based on thermodynamic conditions at the inlet alone, when the flow is choked, or at the outlet as well, when the flow is nonchoked, are used to compute the nozzle flow. These correlations will be discussed in detail in Section 7.4.10.2.

Turbine efficiency is based on losses to isentropic expansion, and shaft work is then calculated using quasi-empirical correlations for stage efficiency. Stage efficiency is affected by many loss factors, such as rotation loss, moisture loss, nozzle-end loss, etc., but only rotation loss and moisture loss are included in the current

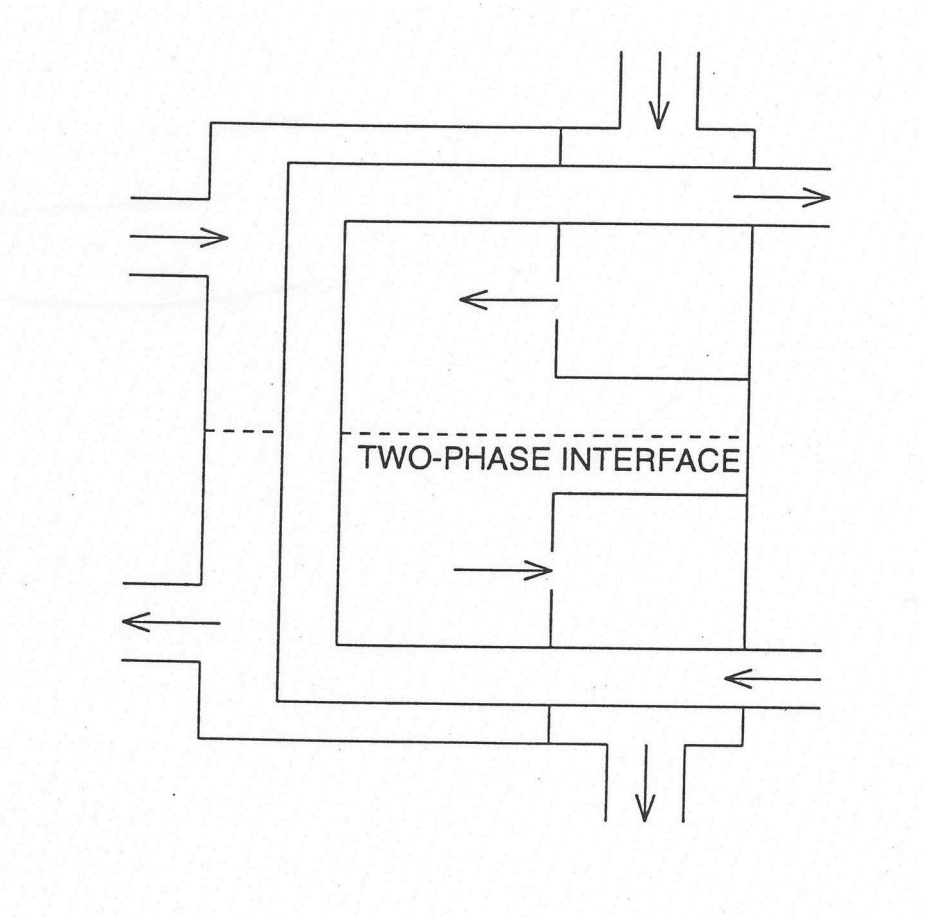


Figure 7.4.11: Desuperheating/Drain Cooler

model because they are significant losses and their functional formulations are known. More loss factors will be included in future model improvements.

7.4.10.2 Analytical Equations

The assumption that the liquid and vapor are strictly separated in the heater models reported above does not apply to the turbine model. Instead, perfect mixing within the compressible volumes which model the turbine stages is now adopted due to the movement of the rotating blades; this is the same assumption made for the energy equation governing the compressible volumes discussed in Section 7.2. Both the mass and the energy equations used in the turbine model are the same as those used in the heater models and in Section 7.2 except that the energy equation used in the turbine model has an additional term included accounting for the energy loss through the work done by the turbine.

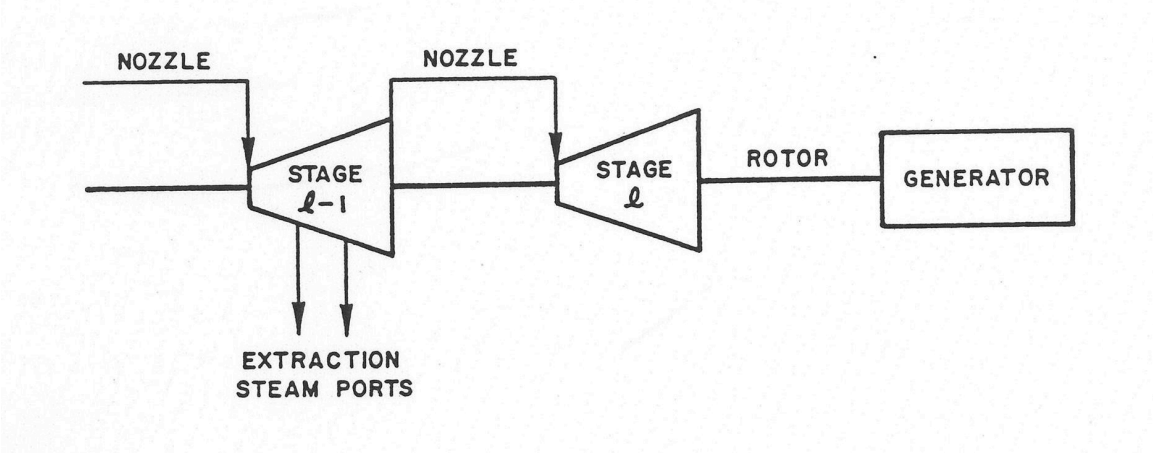


Figure 7.4.12: Turbine

For simplicity, look first at the energy equation used in the heater models,

$$\frac{\partial \rho h}{\partial t} = -\frac{\partial \rho h u}{\partial z} - \frac{\partial q}{\partial z} + \frac{\partial P}{\partial t} \tag{7.4-29}$$

which is simply the same equation as Eq. (7.2-6). Then by integrating this equation over the entire volume V_l for each volume l and writing a separate energy equation for each volume with the work term added, Eq. (7.4-29) becomes

$$\frac{d\left(m_l h_l\right)}{dt} = \sum_j h_j w_j \text{sgn}\left(j,l\right) + Q_l + V_l \frac{dP_l}{dt} - H_l \tag{7.4-30}$$

when the work term H_l is in terms of turbine efficiency based on losses to isentropic expansion and is expressed as

$$H_l = \eta_l w_{in} \Delta \overline{h}_l \tag{7.4-31}$$

where η_l is the stage efficiency, w_{in} is the flow rate at the inlet nozzle, and $\Delta \overline{h}_l$is the loss due to isentropic expansion, defined as

$$\Delta \overline{h}_l = h_{l-1}\left(P_{l-1}, s_{l-1}\right) - \overline{h}_l\left(P_l, S_{l-1}\right) \tag{7.4-32}$$

Note that $\overline{h}_l$ is evaluated at $P = P_l$ and $s = s_{l-1}$, which is the entropy at the previous stage $l-1$.

Fundamentals of the turbine thermodynamics are given in many standard books and Salisbury (Ref. *7-10*) discusses the subject comprehensively.

The stage efficiency η_l is just the blade efficiency $\eta_{l,B}$ minus loss factors from rotation $(R_{l,R})$ and moisture $(R_{l,m})$,

$$\eta_l = \eta_{l,B} - R_{l,R} - R_{l,M} - R_{misc}` \tag{7.4-33}$$

where R_{misc} represents all other minor losses currently not considered. In addition, an exhaust loss term has to be included in the right-hand side of Eq. (7.4-33) if the turbine stage is the last stage of the turbine. The exhaust loss is expressed as

$$R_E = K_E V_{exit}$$

where V_{exit} is the velocity leaving the last stage and K_E is the exhaust loss constant.

The blade efficiency is defined as the ratio of the energy transfer to the blades to the theoretically available energy at the nozzle. From the velocity vector diagram shown in Figure 7.4.13 (the variables in the figure are explained in the context), the rate of energy transfer from the steam to the blades is obtained as the product of the blade velocity V_B and the force F exerted by the steam on the blades. The available energy (rate) at the nozzle is simply the kinetic energy in the isentropic expansion of the steam. Therefore,

$$\eta_{l,B} = \frac{F V_B}{w_{in} \frac{V_o^2}{2}} \tag{7.4-34}$$

where *win* is the same as in Eq. (7.4-31) and V_o is the theoretical nozzle (steam) velocity in the isentropic expansion. By the principle of impulse and momentum, the force on the blades is equal to the steam flow rate w_{in} multiplied by the total change in steam velocity, relative to the blades, i.e.,

$$F = w\left(V_a + V_b\right) \tag{7.4-35}$$

where V_a and V_b are the steam velocity relative to the blade and parallel to the motion of the blade at the blade entrance and exit, respectively, as indicated in Figure 7.4.13. The plus sign in Eq. (7.4-35) indicates, as a result of the vector operation, that V_a and V_b are in opposite directions. Substituting Eq. (7.4-35) into Eq. (7.4-34) thus gives the blade efficiency as

$$\eta_{l,B} = \frac{w_{in}\left(V_a + V_b\right) V_B}{w_{in} \frac{V_o^2}{2}} = \frac{2\left(V_a + V_b\right) V_B}{V_o^2} \tag{7.4-36}$$

The actual velocity V_1 of steam leaving the nozzle is slightly less than V_o and becomes even smaller when the effect of reaction, i.e., energy released in the bucket (Ref. *7-10*), is considered. Thus,

$$V_1 = V_o C_n a \tag{7.4-37}$$

with $a \equiv (1-x)^{1/2}$, C_n the nozzle velocity coefficient, and x the fraction of stage energy released in the bucket system.

In an actual turbine system, it is impossible for the entrance angle of the steam to the blades system to be horizontal or in the plane of the wheel; otherwise, there would be mechanical interference between the rotating blades and the nozzle. Therefore, the steam has to leave the nozzle at a nozzle angle α, as shown in Figure 7.4.13. Furthermore, it is impossible for the steam to exit the blades at an angle of zero with respect to the plane of rotation of the blades; otherwise, there would be no way for the flow of steam to progress axially through the turbine. Thus, the blade exit angle γ in Figure 7.4.13 provides an axial leaving velocity component perpendicular to the plane of rotation of the blades for the exit velocity V_3.

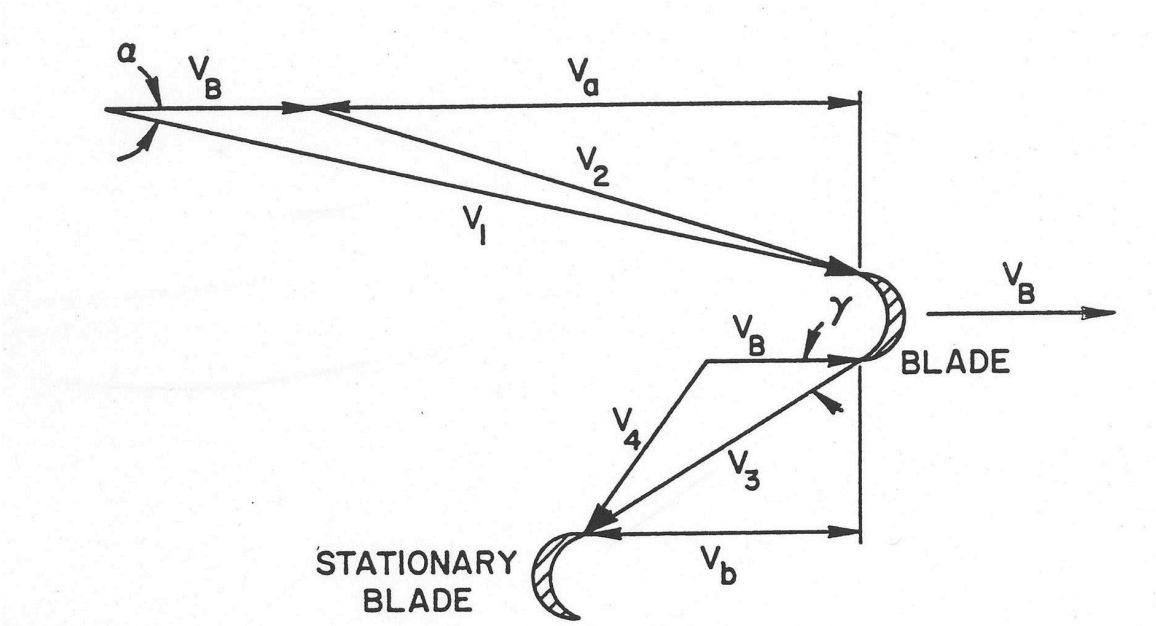


Figure 7.4.13: Velocity Vector Diagram for Single Row of Moving Blades

Two additional loss factors must also be considered in a real turbine system: the fraction of the kinetic energy entering the blades which is realized at the blade exit C_b^2 and the fraction of the stage energy released in the blades which is finally realized at the bucket exit, C_r^2. The exit velocity can then be represented (Ref. *7-10*) by

$$V_3 = \left(C_b^2 V_2^2 + C_r^2 x V_o^2\right)^{1/2} \tag{7.4-38}$$

in a single row of moving blades. For simplicity, C_b will be called the bucket coefficient, C_r the reaction coefficient, and x the reaction fraction. V_3 is in fact the steam velocity at the blade exit, relative to the blade, while V_2 is the steam velocity at the blade entrance, relative to the blade, and, by the law of cosines, is written as,

$$V_2 = \left(V_1^2 + V_B^2 - 2V_1 V_B \cos\alpha\right)^{1/2} \tag{7.4-39}$$

The absolute steam velocity at the blade exit, V_4, is then

$$V_4 = \left(V_3^2 + V_B^2 - 2V_3 V_B \cos\gamma\right)^{1/2} \tag{7.4-40}$$

In summary, by using the fact that

$$V_a = V_1 \cos\alpha - V_B$$

and

$$V_B = V_3 \cos\gamma$$

the blade efficiency in Eq. (7.4-36) can be rewritten explicitly, along with Eq. (7.4-37) to Eq. (7.4-39), as

$$\eta_{l,B} = \frac{2V_B}{V_o}\left\{aC_n\cos\alpha - \frac{V_B}{V_o} + \cos\gamma\left[C_r^2 x + C_b^2 C_n^2 a^2 \left(1 + \left(\frac{V_B}{V_o}\right)^2 \frac{1}{C_n^2 a^2} - 2\frac{V_B}{V_o}\frac{1}{C_n a}\cos\alpha\right)\right]^{1/2}\right\} \tag{7.4-41}$$

Salisbury recommends the following expressions for the rotation loss and moisture loss:

$$R_{l,R} = K_{l,R}\frac{\left(V_h/V_o\right)^3}{A_{l,n}}$$

and

$$R_{l,M} = K_{l,M}\left(1 - x_{l-1}\right)$$

where $K_{l,R}$ is the rotation loss constant, $K_{l,M}$ is the moisture loss constant, $A_{l,n}$ is the nozzle area, and x_{l-1} is the fluid quality at the previous stage $l-1$. All the quantities used in Eq. (7.4-41) are volume dependent.

Since the work of the turbine is done through the rotors, which are aligned along a shaft to drive the generator, a relation has to be established between the turbine work and the rotor angular velocity ω. The turbine work is a summation of the work by individual turbine stages, and the turbine work is related to the rotor angular velocity as,

$$\sum_l H_l = \omega\tau_T \tag{7.4-42}$$

where τ_T is the turbine torque. The rotor angular velocity is changing during a transient according to the relative magnitudes of the turbine torque and the generator torque as follows,

$$I\frac{d\omega}{dt} = \tau_T - \tau_G \tag{7.4-43}$$

where I is the turbine/generator rotor moment of inertia and τ_G is the generator torque.

As mentioned in Section 7.4.10.1, the momentum equation used in the nozzle model is completely different from the one used in the heater models and in Section 7.2 because of the compressible flow considered for the nozzle. For nonchoked flow, the nozzle velocity is based on thermodynamic conditions at the inlet and at the outlet and is calculated from

$$V_{nz} = V_o C_n = \sqrt{2}C_n\Delta\overline{h}_l^{1/2} \tag{7.4-44}$$

where C_n is just the nozzle coefficient and $\Delta\overline{h}_l$ is defined in Eq. (7.4-32), and the mass flux G is given by

$$G = V_{nz}\overline{\rho}_l \tag{7.4-45}$$

where $\overline{\rho}_l$ is the density evaluated at P_l and $\overline{h}_l$, i.e., $\overline{\rho}_l = \overline{\rho}_l\left(P_l, \overline{h}_l\right)$ For choked flow, the nozzle velocity is based on thermodynamic conditions at the inlet only and is determined (Refs. *7-11*, *7-12*) separately for steam flow and for two-phase flow as:

for steam:

$$V_{nz} = V_o C_n = -0.6611 C_n\left[\frac{P_{l-1}}{\rho_{l-1}}\right]^{1/2} \tag{7.4-46}$$

when the criterion $0.545P_{l-1} \geq P_l$ is satisfied,

for two-phase flow:

$$V_{nz} = V_o C_n = \sqrt{2}C_n\left[\frac{P_{l-1} - P^*}{\rho_{l-1}}\right]^{1/2} \tag{7.4-47}$$

where

$$P^* = 0.284P_{sat}\left(T_{l-1}\right)\left(\frac{-0.284f\left(T_{l-1}\right)}{f\left(T_r\right)} + 1\right)$$
$$f\left(T\right) = 113.368 - 0.14T \tag{7.4-48}$$

and

$$T_r = 467.37K$$

when the criterion $P^* \geq P_l$ is satisfied. For both steam flow and two-phase flow, the mass flux G is calculated by

$$G = V_{nz}\rho_{l-1} \tag{7.4-49}$$

Where ρ_{l-1} is the density at the previous stage.

In the steady state, each of the user-specified nozzle flow rates is compared to the calculated nozzle flow rate, which is the nozzle velocity, computed (according to the steady-state pressure conditions) from one of the equations in Eq. (7.4-44), Eq. (7.4-46), and Eq. (7.4-47), multiplied by the user-specified nozzle area. If the calculated nozzle flow rate differs from the user-specified nozzle flow rate, the computed nozzle velocity together with the user-specified nozzle flow rate is used to adjust the user-specified nozzle area, so that the two nozzle flow rates (computed and user-input) will then be consistent. The adjusted nozzle area in the steady state is used for calculations throughout the transient. This flow area calibration algorithm is also used in the relief valve model, discussed in Section 7.4.11, to modify the user-input relief valve flow area based on the user-input relief valve capacity.

7.4.10.3 Discretized Equations

The energy equation for the turbine stage, Eq. (7.4-30), contains only an additional term, $-H_l$, compared with the energy equations for the heater models or for non-heater compressible volumes Eq. (7.2-26). The work term H_l is written separately to emphasize the energy loss of the steam during isentropic expansion while driving the blades in the turbine stages; alternatively, H_l can be absorbed into the heat source term Q_l in Eq. (7.4-30), since it is only an energy term to be subtracted from the energy equation. Eq. (7.4-30) can then be rewritten as,

$$\frac{d\left(m_l h_l\right)}{dt} = \sum_j h_j w_j \text{sgn}\left(j,l\right) + Q_l + V_l \frac{dP_l}{dt} \tag{7.4-50}$$

where Q_l now includes implicitly an energy loss due to the stage work. Eq. (7.4-50) has the same form as Eq. (7.2-26), and so Eq. (7.4-50) can be discretized in exactly the same way as Eq. (7.2-26); see Section 7.2 for details of the discretization.

The discretization of the momentum equation for nozzles is a complicated process, since the momentum equation now is entirely different from the one used in the previous models. In the existing solution scheme of the balance-of-plant coding, the momentum Eq. (7.2-5) is rewritten, after discretization, such that the change in mass flow rate in each segment is expressed in terms of the changes in volume pressures at the segment inlet and outlet as,

$$\Delta W = \frac{a_1 + \left[a_2 + \Delta t\left(\Delta P_{in} - \Delta P_{out}\right)\right]}{a_o - a_3} \tag{7.4-51}$$

Then an $L \times L$ matrix equation is created by combining Eq. (7.4-51) and the discretized energy equation to solve for all L volume pressure changes simultaneously. In order to integrate the momentum equations for nozzles into the established matrix equation solution scheme, Eq. (7.4-44) to Eq. (7.4-49) have to be discretized to a form, similar to Eq. (7.4-51), relating flow changes to pressure changes in the volumes at the segment ends.

The derivation will begin with the momentum equation for nonchoked flow and proceed to choked flow, for which two separate treatments are needed for single-phase (steam) and two-phase flow.

Consider the flow rate w_{in} at the inlet nozzle to stage l as shown in Figure 7.4.12. By definition, the flow rate for nonchoked flow is the mass flux (or mass velocity) G in Eq. (7.4-45) multiplied by the nozzle area A_n; for brevity, w_{in} will be designated as w, V_{nz} as V, and A_n as A in the derivation. Thus,

$$w = \overline{\rho}_l V A \tag{7.4-52}$$

Eq. (7.4-52) can be written in a differential form as,

$$dw = A\overline{\rho}_l dV + AV d\overline{\rho}_l \tag{7.4-53}$$

where dV can be obtained from Eq. (7.4-44),

$$dV = \frac{\sqrt{2}}{2} C_n \Delta \overline{h}_l^{-1/2} \left(dh_{l-1} - d\overline{h}_l \right) = \frac{V}{2\Delta \overline{h}_l} \left(dh_{l-1} - d\overline{h}_l \right) \tag{7.4-54}$$

and $d\overline{\rho}_l$ can be expressed as

$$d\overline{\rho}_l = -\overline{\rho}_l^2 d\overline{v}_l \tag{7.4-55}$$

with $1/\overline{\rho}_l = \overline{v}_l \left(P_l, \overline{h}_l \right)$ which follows the definition of $\overline{\rho}_l$ in Eq. (7.4-45). Substituting Eq. (7.4-54) and Eq. (7.4-55) into Eq. (7.4-53) yields, after rearranging,

$$dw = \frac{w}{2\Delta \overline{h}_l} \left(dh_{l-1} - d\overline{h}_l \right) - w\overline{\rho}_l d\overline{v}_l \tag{7.4-56}$$

Both $d\overline{h}_l$ and $d\overline{v}_l$ in Eq. (7.4-56) are related to the pressure of stage l as, respectively,

$$d\overline{h}_l = \left. \frac{\partial \overline{h}_l}{\partial P} \right|_{s_{l-1}} dP_l + \left. \frac{\partial \overline{h}_l}{\partial s} \right|_{P_l} ds_{l-1} \tag{7.4-57}$$

(remember, $\overline{h}_l \equiv \overline{h}_l \left(P_l, s_{l-1} \right)$ as seen in Eq. (7.4-32), and

$$d\overline{v}_l = \left. \frac{\partial \overline{v}_l}{\partial P} \right|_{\overline{h}_l} dP_l + \left. \frac{\partial \overline{v}_l}{\partial h} \right|_{P_l} d\overline{h}_l \tag{7.4-58}$$

By combining Eq. (7.4-56), Eq. (7.4-57), and Eq. (7.4-58), the finite change in flow rate can be rewritten as

$$\begin{aligned} \Delta w = {} & \frac{w}{2\Delta \overline{h}_l} \left(\Delta h_{l-1} - \left. \frac{\partial \overline{h}_l}{\partial P} \right|_{s_{l-1}} \Delta P_l - \left. \frac{\partial \overline{h}_l}{\partial s} \right|_{P_l} \Delta s_{l-1} \right) \\ & - w\overline{\rho}_l \left[\left. \frac{\partial \overline{v}_l}{\partial P} \right|_{\overline{h}_l} \Delta P_l + \left. \frac{\partial \overline{v}}{\partial h} \right|_{P_l} \left(\left. \frac{\partial \overline{h}_l}{\partial P} \right|_{s_{l-1}} \Delta P_l + \left. \frac{\partial \overline{h}_l}{\partial s} \right|_{P_l} \Delta s_{l-1} \right) \right] \end{aligned} \tag{7.4-59}$$

At this point, it is assumed that the change in enthalpy in the previous stage Δh_{l-1} can be expressed explicitly as a function of the change in pressure in stage $l-1$ and the current time flows in the attached segments, without introducing unacceptable errors; that is, Eq. (7.2-33) can be simplified for the turbine stages (compressible volumes) by neglecting Δw_j, giving

$$\begin{aligned} \Delta h_l = \frac{v_l}{V_l} \Delta t \left\{ -h_l \sum_j w_j \text{sgn}\left(j,l\right) + \sum_j h_j w_j \text{sgn}\left(j,l\right) + Q_l \right\} + v_l \Delta P_l \\ \equiv \{j,l\} + v_l \Delta P_l \end{aligned} \tag{7.4-60}$$

Treating the stage enthalpy implicitly would result in a discretization form which is tediously complicated and cumbersome; however, this might be included in future model improvements.

The final discretization form is arrived at by replacing the subscript l with $l-1$ in Eq. (7.4-60) and then using Eq. (7.4-60) in Eq. (7.4-59) to give

$$\begin{aligned} \Delta w = {} & \frac{w}{2\Delta \overline{h}_l} v_{l-1} \Delta P_{l-1} - w \left[\left(\frac{1}{2\Delta \overline{h}_l} + \overline{\rho}_l \left. \frac{\partial \overline{v}_l}{\partial h} \right|_{P_l} \right) \left. \frac{\partial h_l}{\partial P} \right|_{s_{l-1}} + \overline{\rho}_l \left. \frac{\partial \overline{v}_l}{\partial P} \right|_{\overline{h}_l} \right] \Delta P_l \\ & + w \left[\frac{1}{2\Delta \overline{h}_l} \{l-1, j\} - \left(\frac{1}{2\Delta \overline{h}_l} + \overline{\rho}_l \left. \frac{\partial \overline{v}_l}{\partial h} \right|_{P_l} \right) \left. \frac{\partial \overline{h}_l}{\partial s} \right|_{P_l} \Delta s_{l-1} \right] \end{aligned} \tag{7.4-61}$$

The two discretized momentum equations, Eq. (7.4-51) and Eq. (7.4-61) have analogous forms, as can be seen by comparing terms; for instance, $\Delta t \;/\; [a_o - a_3]$ in Eq. (7.4-51) corresponds to $wv_{l-1}/\left(2\Delta\overline{h}_l\right)$ in Eq. (7.4-61), and so on. Thus, when considering the volume pressure change due to the contribution of a nozzle segment, Eq. (7.4-61) becomes the counterpart of Eq. (7.4-51). Eq. (7.4-61) can be combined with the discretized energy Eq. (7.2-47) to render Eq. (7.2-55), which is expressed entirely in terms of changes in compressible volume pressures and is used to form the $L \times L$ matrix equation system.

Since a functional form expressing enthalpy as a function of pressure and entropy is not available, the known expression for entropy as a function of pressure and enthalpy is used to compute the derivative terms $\left(\partial\overline{h}_l/\partial P\right)$ and $\left(\partial\overline{h}_l/\partial s\right)$ in Eq. (7.4-61). In the case of $\left(\partial\overline{h}_l/\partial P\right)$ at $s = s_{l-1}$, the derivative is calculated by using the known values of $\overline{h}_l$ and P_l as the starting points and then iteratively searching for the other enthalpy value at pressure P_l plus a small increment and at entropy s_{l-1}. The derivative $\left(\partial\overline{h}_l/\partial s\right)_{P_l}$ is easier to obtain from

$$\left.\frac{\partial\overline{h}_l}{\partial s}\right|_{P_l} = 1 \Bigg/ \left.\frac{\partial\overline{s}_{l-1}}{\partial h}\right|_{P_l} \tag{7.4-62}$$

since entropy is a known function of pressure and enthalpy. In addition, the change in the entropy Δs_{l-1} is also computed explicitly.

An alternative to calculating Δh_{l-1} in Eq. (7.4-59) is to express Δh_{l-1} in terms of the pressure and the entropy, i.e.,

$$\Delta h_{l-1} = \left.\frac{\partial h_{l-1}}{\partial P}\right|_{s_{l-1}} \Delta P_{l-1} + \left.\frac{\partial h_{l-1}}{\partial s}\right|_{P_{l-1}} \Delta s_{l-1} \tag{7.4-63}$$

Since $h_{l-1} = h_{l-1}\left(P_{l-1}, s_{l-1}\right)$. Then, the discretization of the momentum equation becomes

$$\begin{aligned}\Delta w = {} & \frac{w}{2\Delta\overline{h}_l}\left.\frac{\partial h_{l-1}}{\partial P}\right|_{s_{l-1}} \Delta P_{l-1} - w\left[\left(\frac{1}{2\Delta\overline{h}_l} + \overline{\rho}_l\left.\frac{\partial\overline{v}_l}{\partial h}\right|_{P_l}\right)\left.\frac{\partial\overline{h}_l}{\partial h}\right|_{s_{l-1}} + \overline{\rho}_l\left.\frac{\partial\overline{v}_l}{\partial P}\right|_{\overline{h}_l}\right]\Delta P_l \\ & + w\left[\frac{1}{2\Delta\overline{h}_l}\left.\frac{\partial h_{l-1}}{\partial s}\right|_{P_{l-1}} - \left(\frac{1}{2\Delta\overline{h}_l} + \overline{\rho}_l\left.\frac{\partial\overline{v}_l}{\partial h}\right|_{P_l}\right)\left.\frac{\partial\overline{h}_l}{\partial s}\right|_{P_l}\right]\Delta s_{l-1}\end{aligned} \tag{7.4-64}$$

The calculation of $(\partial h_{l-1}/\partial s)$ at $P = P_{l-1}$ (using an expression of the same form as Eq. (7.4-62)) in Eq. (7.4-64) and the calculation of the term $\{l-1, j\}$ in Eq. (7.4-61) probably take about the same amount of time. However, Eq. (7.4-64) is more time-consuming to solve than Eq. (7.4-61) because of the need, to use the equation of entropy as a function of pressure and enthalpy, to evaluate $(\partial h_{l-1}/\partial P)$ at $s = s_{l-1}$ by iteration. In addition, Eq. (7.4-61) can be improved by expressing $\{l-1, j\}$ in an implicit form, whereas the additional Δs_{l-1} term makes Eq. (7.4-64) more explicit (note the difference in the Δs_{l-1} terms between the two equations). Currently, Eq. (7.4-61) is used in the turbine model.

For choked flow of steam, the flow rate at the inlet nozzle to stage l is obtained by multiplying Eq. (7.4-49) by the nozzle area,

$$w = \rho_{l-1} V A \tag{7.4-65}$$

where the nozzle velocity V is given in Eq. (7.4-46). In the following derivations for choked flow of steam, the subscript $l = 1$ is omitted for convenience, since the nozzle velocity is based on thermodynamic conditions at the inlet stage only. Differentiating Eq. (7.4-65) and rearranging gives,

$$dw = \frac{w}{2}\left(\frac{dP}{P} + \frac{d\rho}{\rho}\right) \tag{7.4-66}$$

where, from Eq. (7.4-55) and Eq. (7.4-58),

$$d\rho = -\rho^2\left(\left.\frac{\partial v}{\partial P}\right|_h dP + \left.\frac{\partial v}{\partial h}\right|_P dh\right) \tag{7.4-67}$$

Now, introducing Eq. (7.4-67) into Eq. (7.4-66) and rewriting the differential change in flow in a finite change form yields,

$$\Delta w = \frac{w}{2}\left[\frac{\Delta P}{P} - \rho\left(\left.\frac{\partial v}{\partial P}\right|_h \Delta P + \left.\frac{\partial v}{\partial h}\right|_P \Delta h\right)\right] \tag{7.4-68}$$

where Δh is again given by Eq. (7.4-60) except that the subscript l should now be $l-1$. Thus the discretization of the momentum equation results in, after inserting Eq. (7.4-60) into Eq. (7.4-68) and rearranging,

$$\Delta w = \frac{w}{2}\left[\frac{1}{P} - \rho\left(\left.\frac{\partial v}{\partial P}\right|_h - \left.\frac{\partial v}{\partial h}\right|_P\right)\Delta P - \rho\left.\frac{\partial v}{\partial h}\right|_P \{l-1, j\}\right] \tag{7.4-69}$$

Similarly, Δh in Eq. (7.4-68) can also be expressed in an implicit form as described in Eq. (7.4-64). It is not surprising that ΔP_l does not appear in Eq. (7.4-69), since the flow rate is affected by the volume pressure only at the previous stage as long as the flow is choked.

As can be seen in Eq. (7.4-47) to Eq. (7.4-49), the two-phase choked flow between stages $l = 1$ and l is also dependent on thermodynamic conditions at the inlet turbine stage $l = 1$ only. For simplicity, the subscript $l = 1$ is not shown in the derivation of the equation for change in mass flow rate. It facilitates the derivation if Eq. (7.4-48) is rewritten in a simplified form, by carrying out the algebra in the equation, as,

$$P^* = P_{sat}(T)(a + bT) = P(a + bT) \tag{7.4-70}$$

where $a = 9.325 \times 10^{-2}$ and $b = 2.356 \times 10^{-4}$. The flow rate in the nozzle now is just $w = \rho V A$ with V given by Eq. (7.4-47). From the analogy between Eq. (7.4-46) and Eq. (7.4-47), the differential change in the two-phase choked flow can be deduced readily from Eq. (7.4-66), which is for choked flow for steam, to be,

$$dw = \frac{w}{2}\left(\frac{dP - dP^*}{P - P^*} + \frac{d\rho}{\rho}\right) \tag{7.4-71}$$

From Eq. (7.4-40), it is obvious that

$$dP^* = (a + bT)\,dP + bP dT \tag{7.4-72}$$

where, since the saturation temperature is determined solely by the saturation pressure, dT can be rewritten as,

$$dT = \frac{dT}{dP} dP \tag{7.4-73}$$

Combining Eq. (7.4-71), Eq. (7.4-72), and Eq. (7.4-73) and rearranging results in,

$$dw = \frac{w}{2}\left[\left(\frac{1}{P} + c\frac{dT}{dP}\right)dP - \rho dv\right] \tag{7.4-74}$$

where

$$c = \frac{-b}{1 - a - bT}$$

Since the specific volume in a two-phase fluid is a function of pressure and quality, dv in Eq. (7.4-74) should be expressed as,

$$\begin{aligned} dv &= \left[\frac{dv_f}{dP} + x\frac{dv_{fg}}{dP} - \left(\frac{dh_f}{dP} + x\frac{dh_{fg}}{dP}\right)\frac{v_{fg}}{h_{fg}}\right]dP + \frac{f_{fg}}{h_{fg}}dh \\ &\equiv [v, h]\,dP + \frac{v_{fg}}{h_{fg}}dh \end{aligned} \tag{7.4-75}$$

Substituting Eq. (7.4-75) into Eq. (7.4-74) gives the finite change in the nozzle flow

$$\Delta w = \frac{w}{2}\left[\left(\frac{1}{P} + c\frac{dT}{dP} - \rho\left[v,h\right]\right)\Delta P - \rho\frac{v_{fg}}{h_{fg}}\Delta h\right] \tag{7.4-76}$$

Finally, the discretized equation for two-phase choked flow is reached by applying Eq. (7.4-60) to Eq. (7.4-76),

$$\Delta w = \frac{w}{2}\left[\left(\frac{1}{P} + c\frac{dT}{dP} - \rho\left[v,h\right] - \frac{v_{fg}}{h_{fg}}\right)\Delta P - \rho\frac{v_{fg}}{h_{fg}}\left\{l-1,j\right\}\right] \tag{7.4-77}$$

dT/dP in Eq. (7.4-77) is computed by introducing perturbations in the pressure in the following equation (Ref. *7-13*),

$$T = c_1 + \frac{c_2}{\ln P + c_3} \tag{7.4-78}$$

where $c_1 = 0.426776 \times 10^2$, $c_2 = -0.389270 \times 10^4$ and $c_3 = -0.948654 \times 10^1$ if $0.000611 \leq P \leq 12.33$ MPa, or $c_1 = -0.387592 \times 10^3$, $c_2 = -0.125875 \times 10^5$, and $c_3 = -0.152578 \times 10^2$ if $12.33 \leq P < 22.1$ MPa. Eq. (7.4-77) does not contain ΔP_l as in the discretized equation for choked flow for steam, Eq. (7.4-69).

7.4.11 Relief Valve

A relief valve is part of the System Pressure Relief System used to relieve overpressure transients occurring during plant isolations and load rejections. The relief valves open rapidly (self-actuated) during plant transients to discharge fluids to the environment or relief tanks and close following the transients so that normal operation can be resumed. Normally the relief valves are located on the main steam line piping, and each valve is piped through its own uniform diameter discharge line.

7.4.11.1 Model Description

Relief valves, like the other types of valves used in the balance of plant, are modeled as flow elements, since a relief valve primarily affects mass flow rate and pressure drop along a flow path and thus is best described through the momentum equation. The details of the relief valve model are dictated by the behavior of this type of valve. A relief valve is normally closed, opening very quickly when the pressure drop across the valve reaches a specified set pressure. The fractional valve opening area is related to the pressure drop across the valve by a hysteresis curve; the one used in the current model is shown in Figure 7.4.14. Valve flow area varies linearly from fully open at or above the accumulated pressure to partially closed to the fraction A(1) at the set pressure. The flow area then remains constant until the valve pressure drop is below the blowdown pressure, at which point the valve closes fully. The relief valve, therefore, behaves in many respects like a check valve. However, flow through the relief valve is normally choked and so must be modeled using a momentum equation such as the ones described in Section 7.4.10 for choked flow in the nozzle model. The relief valve is therefore modeled as a nozzle which can open and shut in much the same way as the check valve does.

In order to avoid the numerical instabilities which might be caused by the step changes in flow area shown at the blowdown and set pressures in Figure 7.4.14, the step changes are modified to linear changes of flow area as a function of pressure within a response time. This alteration of the ideal hysteresis curve of Figure 7.4.14 actually makes the model more realistic, since an actual relief valve exhibits a response time which is the maximum valve opening time (for example, 0.2 second). This response time is to be specified by the user as part of the valve input data.

In summary, the relief valve model consists of a modified check valve model and a modified nozzle model. The modified check valve model determines the fractional valve opening area, according to the hysteresis curve and the pressure difference across the relief valve, and then the modified nozzle model uses this flow area in calculating the flow rate through the relief valve.

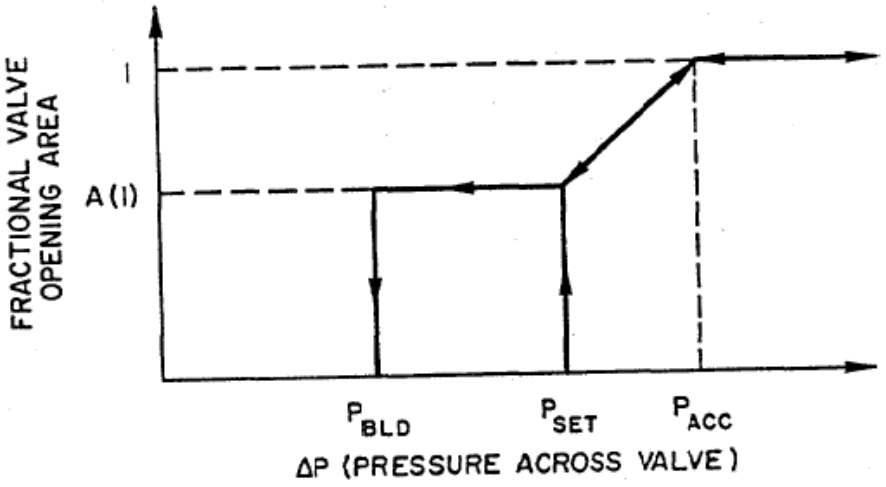


Figure 7.4.14: Simple Relief Valve Hysteresis Curve

The difference between the check valve model and the modified check valve model is that the check valve changes state (close or open) by adjusting the orifice coefficient in the incompressible flow momentum equation according to a user-input table of orifice coefficients versus time, whereas the modified check valve model changes state by adjusting the fractional valve opening area to a user-input value, in accordance with the hysteresis curve, within a user-input response time. In fact, the orifice coefficient functions like an inverse fractional opening area, since the orifice coefficient is inversely proportional to the square of the flow rate through the check valve under a fixed pressure drop across the valve Eq. (7.2-24). The modified nozzle model differs from the nozzle model in that there is sometimes no flow through the nozzle (the relief valve) if the fractional valve opening area generated by the modified check valve model is zero, whereas the nozzle model always has a flow, either choked or nonchoked, between two stages.

7.4.11.2 Analytical Equations and Discretized Equations

All aspects discussed in Section 7.4.10.2 and Section 7.4.10.3 about the momentum equations for the nozzle model apply to the relief valve model. Two additional points to note are that 1) the work term H_l in Eq. (7.4-30) and Eq. (7.4-50) (included in Q_l) is removed when these two energy equations are used for the volume following the relief valve, since there is no work done by the flow through the valve and 2) when there is no flow through the valve, the pressure change in the volumes at either end of the relief valve will have no contributions from the momentum equation of the relief valve (as is easily seen in Eq. (7.4-61), Eq. (7.4-69), and Eq. (7.4-77) when $w = 0$).

7.4.12 Steady-State Initialization

The steady-state initialization process takes the input data entered by the user and from the data computes the steady-state parameters which characterize each component. The elements of the initialization process for the models discussed in this report can be separated into three groups: those for heater models, those for the turbine model, and those for the relief valve model. These three groups of calculations will now be discussed separately.

7.4.12.1 Steady-State Initialization for the Heater Models

Some steady-state parameters must be calculated for all heater models, both open and closed. These include:

1. compute the heater temperature from the equation of state,
2. determine the quality of the heater if the user inputs the two-phase interface and vice versa, by Eq. (7.4-5) or Eq. (7.4-8),
3. check the mass conservation from all of the inlet and outlet flows,
4. calculate the heat source term steady-state value from the net energy entering the volume through the inlet and outlet flows,
5. introduce the pseudo heat conduction coefficient, to account for the minor energy imbalance resulting from inconsistencies between user-specified thermodynamic conditions and the SASSYS-1 correlations for the thermodynamic properties, and
6. calculate the area-weighted quality for the outlet flow, if the two-phase interface intersects any of the outlet openings, from Eq. (7.4-2).

In addition, there are some calculations required only for the initialization of closed heaters. These include:

7. initialize the heat transfer coefficients, both on the tube side and on the shell side,
8. determine the calibration factor for the tube surface heat transfer area,
9. initialize the fluid enthalpy at each node on the tube side,
10. find the temperature distributions of the tube side fluid and the metal tube, and
11. include the heat transfer energy between the shell side fluid and the metal tube in the heat source term steady-state value (item (4) above).

Furthermore, for those heaters which contain a drain and/or a desuperheating region, the following initialization is also performed for the shell side of the drain and/or the desuperheating region:

12. calculate the enthalpy from the equation of state,
13. initialize the heat transfer coefficient, from Eq. (7.4-10),
14. compute the heat source term steady-state value from the net energy gain through the inlet and outlet flows, and
15. determine the calibration factor, separate from the one used for the shell side of the heater, for the tube surface heat transfer area to obtain a balance between the energy transfer through the tube boundary and the heat source term steady-state value from item (14).

7.4.12.2 Steady-State Initialization for the Turbine Model

The turbine model requires initialization of parameters both for each turbine stage and for each nozzle between stages. The job of the turbine stage steady-state initializer is the following:

1. check the mass conservation from all of the in]et and outlet flows,
2. compute the steady-state stage work, from Eq. (7.4-31),
3. calculate the turbine torque and set the generator torque equal to the turbine torque,
4. check the energy conservation (work balanced with the net energy in through flows), and
5. introduce a pseudo-heat conduction coefficient to account for any minor energy imbalance within the stage.

The steady-state nozzle initializer performs the following:

1. calibrate the nozzle area according to the user-specified flow rate, using the mass flux calculated from Eq. (7.4-45) or Eq. (7.4-49), and
2. compute the isentropic enthalpy $\overline{h}_l(P_l, s_{l-1})$ and density $\overline{\rho}_l(P_l, \overline{h}_l)$.

7.4.12.3 Steady-State Initialization for the Relief Valve Model

The steady-state relief valve initializer does the following:

1. calibrate the valve flow area when the valve is fully open, according to the user-specified relief valve capacity (since the relief valve is normally closed in the steady state), using the same equations for calculating the mass flux as are used in the nozzle model, and
2. set the flag to reflect the state of the relief valve as being on the upper section of the hysteresis curve in Figure 7.4.14, or on the lower section of the hysteresis curve.

7.4.13 Code Implementation and Operation

7.4.13.1 The Transient Solution Algorithm

This section describes the incorporation of the component models discussed in this section into the transient solution algorithm discussed in Section 7.2.6.2.

Perhaps the easiest way to document the revised transient solution algorithm is to start with the algorithm of Section 7.2.6.2. The original algorithm is broken up into eleven steps; some of these steps are affected by the addition of the heater, turbine, and relief valve models, and some steps are not. The modifications to the algorithm which are necessary to incorporate the additional models are as follows:

Steps 1 and 2:
These involve only the steam generator and superheater models and thus are not affected by the additional models.

Step 3:
The calculation of the matrix coefficients and source terms performed in this step must properly account for the enthalpy of fluid leaving any of the heater volumes, as discussed in Section 7.4.2.2, *7.4.7.2*, and *7.4.8.2* of this report. Also, the heat source term Q_l^n should include the stage work term H_l^n for each stage of a turbine.

Step 4:
The incompressible flow momentum equation for which the coefficients are computed in this step must be replaced by Eq. (7.4-61), Eq. (7.4-69), or Eq. (7.4-77) (the choice of equation being dependent upon flow conditions, as discussed in Section 7.4.10.3) in the case of a nozzle or relief valve.

Steps 5 and 6:
These involve the mechanics of assembling and solving the matrix equation and so are not affected by the additional models.

Step 7:
In addition to updating the volume pressures and the flows in incompressible flow segments, the flows in any nozzles or relief valves must be updated.

Step 8:
In this step, the heat source terms for any of the heater models discussed in this report are updated. The temperature profiles of the tube side fluid and the metal tube are updated for any closed heaters, and the heater shell side temperature is also updated. In addition, the temperature, enthalpy, and heat source are updated in any drains or desuperheating regions. For any turbine stages, the work term, the turbine torque, and the rotor angular velocity are updated.

Step 9:
Volume-averaged densities and enthalpies are computed here, and this calculation does not change for the new models.

Step 10:
In non-heated flow segments, enthalpy transport is applied at this point to update the enthalpy distribution along each segment. In the heated elements within closed heaters, the fluid enthalpy at each node along the element is updated according to the energy equations discussed earlier. Flow segments which involve a combination of heated and non-heated elements use enthalpy transport in all non-heated elements and compute energy transfer within heated elements.

Step 11:
Pump parameters and gravity heads are computed in this final step, and so this step is not affected by the additional models.

7.4.13.2 Code Operation

The calculation procedures described in Section 7.4.12 and Section 7.4.13.1 are implemented by the set of subroutines and functions listed in Table 7.2.1. Most subroutines with names beginning with SS (Steady State) are called from the steady-state balance-of-plant driver SSBOP (the exception is subroutine SSRVW, which is called from subroutine RENUM, the balance-of-plant input data and nodalization routine). Subroutines with names beginning with TS (Transient State) are called from the driver subroutine WTRDRV.

The structure of the balance-of-plant coding, including the subroutines of Table 7.2.1, is shown in Figure 7.2.1 for the steady-state coding and in Figure 7.2.2 for the transient coding.

As seen in Figure 7.2.1, the balance-of-plant operation starts with subroutine RENUM, which is called from the PRIMAR-4 subroutine SSPRM4. SSPPM4 is an initialization subroutine which has been modified to call the balance-of-plant initialization subroutines also. Subroutine SSRVW is called by RENUM to calibrate the relief valve flow areas once the input data for each relief valve have been entered through RENUM. After the work is completed in RENUM, SSPRM4 calls SSBOP to execute the remainder of the steady-state balance-of-plant initialization. All the calculations necessary to complete the steady-state initialization of the balance-of-plant components are done in SSBOP or in subroutines called by SSBOP. First, SSBOP calls the nozzle initializer, SSNZZL, to compute the isentropic enthalpies and the fluid densities following the isentropic expansion, and to adjust the nozzle areas. Next, SSBOP calls SSTRBN, the turbine initializer, to calculate the steady-state turbine stage work, check mass and energy conservation, and initialize the turbine and generator torques. SSBOP then

calls SSHTRW, the heater initializer, to compute the shell-side temperatures and/or the tube-side temperature profile (along with the heat transfer area calibration factor), calculate the heat source terms (including heat source terms in the drain and/or the desuperheating region, if such regions are present), check mass and energy conservation, and find the pseudo-heat conduction coefficient. SSBOP then goes on to call other initialization subroutines.

With the steady-state initialization completed, the balance-of-plant transient calculation starts with subroutine TSBOP by a call from the PRIMAR-4 subroutine STEPTM (see Figure 7.2.2). TSBOP is the driver for the entire waterside transient calculation, including the steam generator and superheater models. TSBOP is called once each PRIMAR-4 timestep. After updating the balance-of-plant standard valves (which operate on the PRIMAR-4 timestep), TSBOP enters a loop that operates on the balance-of-plant timestep, which is generally smaller than the PRIMAR-4 timestep and is never larger than the PRIMAR-4 timestep. For each balance-of-plant timestep, TSBOP first calls the steam generator (and superheater, if any are present in the system) subroutines, then it calls WTRDRV, the driver subroutine for the remainder of the balance of plant. WTRDRV assembles the matrix equation for the compressible volume pressure changes, solves for the pressure changes, and updates the flow rates and the remaining balance-of-plant parameters. In the process of assembling the matrix equation, WTRDRV must compute, for each flow segment, the contribution of each component in the segment (e.g., pipes, valves, pumps, etc.) to the segment momentum equation. These contributions are calculated, for the different component types, by the six subroutines grouped together in Figure 7.2.2, beginning with SGMOM and ending with TSRVW. The first four subroutines are discussed in Section 7.2; the remaining two are TSNZZL, which calculates the contribution for nozzles, and TSRVW, which calculates the contribution for relief valves. Once the matrix equation has been solved for the compressible volume pressure changes, TSNZZL and TSRVW are called again to update nozzle flow rates and relief valve flow rates, respectively. Since relief valves are normally closed, TSRVW checks to find out first if the relief valve is open, to decide whether to proceed with the generation of the momentum equation coefficients or simply to bypass the calculation if the valve is closed. In updating the relief valve flow, subroutine TSRVW calls REVLW to obtain the new valve flow area and then calculates the new valve flow rate; again, TSRVW will skip the computation of the new flow rate and set the new flow rate to zero if the new valve flow area is found to be zero.

WTRDRV moves on to call TSTRBN, which updates the turbine stage work and the turbine torque. Next, WTRDRV calls TSHTRW, which executes most calculations related to heaters, other than those simulated by the simple heater model in Section 7.2. Subroutine TSHTRW updates the shell-side temperature and/or the tube-side temperature profile and also the heat source terms (including the heat source terms in the drain and/or desuperheating regions, if such regions are present). After these calls, the enthalpy transport calculation along unheated flow elements is performed by a call to TRNSPT, which will also account for enthalpy transport in flow paths following heated elements within heaters. Subroutine ARF is then called, as necessary, to determine the new two-phase interface in heaters configured as cylinders lying on the side.

Once WTRDRV has returned control to TSBOP, the balance-of-plant timestep is updated and TSBOP performs another series of calls to the steam generator and superheater subroutines and to WTRDRV, until the end of the PRIMAR-4 timestep is reached. TSBOP then goes on to perform I/O tasks, then returns control to STEPTM.

7.5 References

7-1. L. F. Moody, “An Approximate Formula for Pipe Friction Factors,” *Mech. Eng.*, vol. 69, no. 12, pp. 1005-1006, December 1947.

7-2. J. R. S. Thom, “Prediction of Pressure Drop During Forced Circulation Boiling of Water,” *Int. J. Heat and Mass Transfer,* vol. 7, pp. 709-724, 1964.

7-3. M. Bein and R. Yahalom, “Dynamic Simulation of an LMFBR Steam Generator,” *Proc. of the Second Power Plant Dynamics, Control and Testing Symposium*, Knoxville, TN, September 1975.

7-4. F. W. Dittus and L. M. K. Boelter, Univ. Calif. Publs. Eng. 2, p. 443, 1930.

7-5. J. R. S. Thom, et al., Symposium on Boiling Heat Transfer in Steam Generating Units and Heat Exchangers, *Inst. Mech. Eng. Proc.* 965-66, vol. 180, part 3C, p. 246.

7-6. A. A. Bishop, et al., "Forced Convection Heat Transfer at High Pressure After the Critical Heat Flux," ASME preprint 65-HT-31, 1965.

7-7. A. A. Bishop, et al., Unpublished information, Westinghouse Electric Corp., 1964.

7-8. W. H. Alliston, et al., Unpublished information, U.S. Department of Energy, 1978.

7-9. J. R. S. Thom, W. M. Walker, T. A. Fallon, and G. F. S. Reising, "Boiling in Subcooled Water During Flow Up Heated Tubes or Annuli," Paper 6 presented at the Symposium on Boiling Heater Transfer in Steam Generating Units and Heat Exchangers held in Manchester, IMechE (London), Sept. 15-16, 1965.

7-10. J. K. Salisbury, Steam Turbines and Their Cycles, Robert E. Krieger Publishing Co., New York, 1974.

7-11. J. W. Murdock and J. M. Bauman, "The Critical Flow Function for Superheated Steam," *J. of Basic Engineering, Transactions of the ASME*, pp. 507-518, September 1964.

7-12. W. G. Steltz, "The Critical and Two-Phase Flow of Steam," *J. of Engineering for Power, Transactions of the ASME*, pp. 145-154, April 1961.

7-13. "New Values for the Viscosity of Water Substance," *Mech. Eng.*, vol. 98, no. 7, p.79, July 1976.

7-14. J. H. McFadden, et al., "RETRAN-02–A Program for Transient Thermal-Hydraulic Analysis of Complex Fluid Flow Systems, Volume 1: Equations and Numerics," NP-1850, Vol. 1, Electric Power Research Institute, May 1981.

7.6 Nomenclature

Symbol	Definition	Units
a	Constant $\left(9.325 \times 10^{-2}\right)$	
a_0, a_1, a_2, a_3	Momentum equation coefficients	
A	Flow area	m^2
b	Constant $\left(2.356 \times 10^{-4}\right)$	
B	Valve damping coefficient	$kg - m/s$
c, c_p	Specific heat	$J/kg - K$
c_b	Bucket coefficient	
c_n	Nozzle velocity coefficient	
c_r	Reaction coefficient	
C	Valve calibration constant	m^2
CF	Calibration factor	
CV	Elevation of heater center	m
C_1, C_2, C_3, C_4	Coefficients in the expression for the Moody friction factor ($C_1 = 0.0055$, $C_2 = 20000$, $C_3 = 106$, and $C_4 = 1/3$)	
D_h, D_H	Hydraulic diameter	m
$\widehat{E}$	Total energy	J/kg
f_l	Liquid friction factor	
f_{tp}	Two-phase friction factor	
F	Force	N
F	Heat flux	W/m^2
F	Pressure loss coefficient	
FL	Fouling resistance	$m^2 - K/W$

continues on next page

Table 7.6.1 – continued from previous page

Symbol	Definition	Units
FR	Friction calibration coefficient	
F_1	Valve driving function	$kg - m/s^2$
g	Gravitational constant	m/s^2
G	Mass flux	$kg/m^2 - s$
G_2	Orifice coefficient	
h	Specific enthalpy	J/kg
hp	Primary side heat transfer coefficient in the simple heater model	$W/m^2 - K$
h_{tot}, H	Total heat transfer coefficient	$W/m^2 - K$
H	Height	m
H	Work	W
I	Moment of inertia	$kg - m^2$
j	Lowermost node in a region	
k	Thermal conductivity	$W/m - K$
k	Valve spring constant	kg/s^2
k_p	Thermal conductivity of the primary side in the simple heater model	$W/m - K$
K	Loss constant	
L	Element length, or distance	m
L_B/D_B	Effective length-to-diameter ratio per bend	
m, M	Mass	kg
n	Index on the discretized time variable	
n	Number of nodes in a region	
N_B	Number of bends along a flow path	
Nu	Nusselt number	
P, P^*	Pressure	Pa
POD	Pitch-to-diameter ratio	
Pr	Prandtl number	
q	Heat flux	$J/s - m^2$
Q	Heat source	J/s
Q	Volumetric heat source	W/m^3
Q	Energy transfer rate	W
r	Radius	m
r_5	Two-phase multiplier	
R	Thermal resistance	K/W
R	Loss factor	
Re	Reynolds number	
s	Distance	m
s	Specific entropy	$J/kg - K$
$sgn\,(j,l)$	+1 if the steady state flow in segment j is entering volume l -1 if the steady state flow in segment j is leaving volume l	
t	Time	s
T	Temperature	K
TP	Elevation of two-phase interface	m
u, v	Fluid velocity	m/s
U_{pseu}	Pseudo-heat conduction coefficient	W/K
v	Specific volume	m^3/kg
V	Volume	m^3
V_a, V_b	Steam velocities relative to the blade and parallel to the blade motion at the blade entrance and exit, respectively	m/s
V_1	Absolute steam velocity at the blade entrance	m/s

continues on next page

Table 7.6.1 – continued from previous page

Symbol	Definition	Units
V_2	Steam velocity relative to the blade at the blade entrance	m/s
V_3	Steam velocity relative to the blade at the blade exit	m/s
V_4	Absolute steam velocity at the blade exit	m/s
w	Mass flow rate	kg/s
WR	Wall resistance on water side	
$WRNA$	Wall resistance on sodium side	
x	Quality	
y	Valve stem position	m
z	Spatial variable	m
Z	Spatial location, region length, or elevation of pipe center	m
Δz	$z_0 - z_I$, the elevation difference between the outlet and the inlet of a flow element	m
α	Void fraction	
α	Nozzle exit angle	rad
β	Volumetric quality	
γ	Blade exit angle	rad
δ	Thickness	m
ε	Element roughness	m
η	Efficiency	
θ	Angle of inclination with respect to the vertical	
μ	Dynamic viscosity	$Pa-s$
v	Specific volume	m^3/kg
ρ	Density	kg/m^3
τ	Viscous stress tensor	Pa
τ	Torque	$N-m$
φ	Valve characteristic	
χ	Quality	
ω	Rotor angular velocity	rad/s
Subscripts		
ACC	Accumulated pressure at which value fully opens	
b	Bottom	
B	Blade	
BLD	Blowdown pressure at which valve closes	
D	Drain, or DNB	
E	Exhaust	
f	Liquid	
fg	Vapor minus liquid	
g	Vapor	
G	Generator	
hf	Location of hf	
hg	Location of hg	
i	Node number, region number, inside	
I, in	Inlet	
j	Segment number	
m	Metal tube, or tube wall	
$misc$	Miscellaneous	
M	Moisture, or mid-wall	
nz	Nozzle	

continues on next page

Table 7.6.1 – continued from previous page

Symbol	Definition	Units
Na	Sodium	
o	Outside, theoretical	
O, out	Outlet	
r	Reference	
R	Rotation	
s	Sodium, shell side, steam drum	
S	Surface	
sat	Saturation	
SC	Subcooled zone	
SG	Steam generator	
SH	Superheated zone	
t	Top, tube side, total	
T	Total, turbine	
TP	Boiling zone	
u	Upper	
w	Water	
l	Volume number	
∞	Ambient	
Superscripts		
k, n	Time step index	
n	Normalized	
u	Unnormalized	
$-$	Isentropic quantity	

7.7 Appendices

7.7.1 Listing of Balance-of-Plant Variables

Many of the variables used in the balance-of-plant subroutines are listed below in alphabetical order, together with a short definition of each variable. All input variables are included in the list and are preceded by an asterisk. In the case of a variable which has a counterpart in the sodium loop coding (e.g, the segment flow), the counterpart variable name is listed in parentheses after the variable definition.

*** ALFANZ(NSEG)**
nozzle angle.

*** APMWHD(20,NPUMPW)**
pump head coefficients and torque coefficients (APMPHD).

*** APRMHT(NHTR)**
the heat transfer area between the primary and secondary sides of a heater.

*** AREAW(NELEW)**
cross-sectional area of an element (AREAEL).

*** ARNAS**
the superheater sodium flow area per tube.

*** AXPMHT(NHTR)**
the flow area on the primary side of a heater.

*** BENDW(NELEW)**
the number of bends in a flow element (BENDNM).

*** CBBKCF(NSEG)**
nozzle bucket coefficient.

*** CHCALB(NCVLVW)**
calibration constant for a check valve.

CHDELT(2,NCVLVW)
CHDELT(1) is the closure time for a check valve.

CHDELT(2) is the opening time for a check valve.

*** CHEPS1(NCVLVW)**
the value of the pressure drop across a check valve or the value of the mass flow rate through the check valve at which the valve begins to close.

*** CHEPS2(NCVLVW)**
the value of the pressure drop across a check valve or the value of the mass flow rate through the check valve at which the valve begins to open.

*** CHPHIW(NCVLVW)**
absolute valve characteristic value for a check valve which is fully open.

CHTIME(NCVLVW)
the time since initiation of closure or of opening of a check valve.

*** CNNZCF(NSEG)**
nozzle velocity coefficient.

*** CONSKl(NSEG)**
nozzle rotation loss coefficient.

*** CONSK2(NSEG)**
nozzle moisture loss coefficient.

*** CONSK3**
turbine exhaust loss coefficient.

*** CRRXCF(NSEG)**
nozzle reactor coefficient.

*** CSAREA(NCVW)**
the effective cross-sectional area of a heater.

*** CSARLW(NCVW)**
the effective cross-sectional area of the drain in a heater containing a drain.

*** CSARUP(NCVW)**
the effective cross-sectional area of the desuperheating region in a heater containing a desuperheating region.

*** CVMLTW(2,NSEGW)**
the multiplicity factors at the entrance (1) and exit (2) of a flow segment (CVLMLT).

*** CVPHIC(10,NCVLVW)**
normalized valve characteristic for a closing check valve as a function of time since the start of valve closure.

*** CVPHIO(10,NCVLVW)**
normalized valve characteristic for an opening check valve as a function of the time since the start of valve opening.

*** CVTIMC(10,NCVLVW)**
time table for CVPHIC.

*** CVTIMO(10,NCVLVW)**
time table for CVPHIO.

*** DEWI**
the evaporator booster tube outer diameter.

*** DEWIS**
the superheater booster tube outer diameter.

*** DEWOS**
the superheater steam tube inner diameter.

*** DHLWW(NCVW)**
the hydraulic diameter of the drain in a heater containing a drain.

*** DHNAS**
the superheater sodium hydraulic diameter per tube.

*** DHPMHT(NHTR)**
the hydraulic diameter of the primary side of a heater.

*** DHSHW(NCVW)**
the hydraulic diameter of the shell side of a heater.

*** DHUPW(NCVW)**
the hydraulic diameter of the desuperheating region in a heater containing a desuperheating region.

*** DHW(NELEW)**
hydraulic diameter of an element (DHELEM).

*** DNSW(NCVW)**
compressible volume density.

*** DOUTS**
the superheater steam tube outer diameter.

*** DPACC(NRVLVW)**
the accumulated pressure drop for a relief valve.

*** DPBLD(NRVLVW)**
the blowdown pressure drop for a relief valve.

DPELW(NELEW)
pressure drop across an element (DPRSEL).

*** DPSET(NRVLVW)**
the set pressure drop for a relief valve.

DTSUBO
previous timestep.

*** FLOWLS**
the relief valve capacity at the accumulated pressure drop.

*** FLOWSS(NSEGW)**
steady state flow in each segment (FLOSSL).

FLOW4(NSEGW)
segment flow at the end of the current PRIMAR time subinterval. At the beginning of the subinterval, flow is given by FLOW3, and at the end of the PRIMAR timestep, it is given by FLOW2 (FLOSL2 FLOSL3, FLOSL4)

*** GAMABL(NSEG)**
turbine blade exit angle.

GRAVW(NELEW)
gravity head in an element (GRAVHD).

*** G2PW(NELEW)**
the orifice coefficient in the momentum equation (G2PRDR).

*** HCVI**
the initial upstream enthalpy for a relief valve.

HCVW(NCVW)
specific enthalpy of the compressible volumes.

*** HEADWR(NPUMPW)**
rated pump head (HEADR).

HEADW2(NPUMPW)
pump head at end of PRIMAR timestep (HEADP2).

HEADW3(NPUMPW)
pump head at beginning of timestep (HEADP3).

HEADW4(NPUMPW)
pump head at end of timestep (HEADP4).

HELEW(NELEW)
enthalpy at an element outlet.

*** HIGHLW(NCVW)**
the height of the drain in a heater containing a drain.

*** HIGHUP(NCVW)**
the height of the desuperheating region in a heater containing a desuperheating region.

*** HTOTO(NHTR)**
the initial value of the total heat transfer coefficient between the primary and secondary sides of a heater.

*** HTRELV(NCVW)**
the elevation of the lowest point of a heater.

*** HTRRAD(NCVW)**
the radius of a heater.

*** IBOPRT**
the number of PRIMAR timesteps between full balance-of-plant prints.

*** ICHVLK(2,NCHVLV)**
a flag for the criteria which trigger a check valve to open and to close. ICHVLK(1,IVLV) flags whether an open check valve will start to close based on a pressure criterion (ICHVLK = 1) or on a flow criterion (ICHVLK = 2). ICHVLK(2,IVLV) does the same for the opening criteria of a closed valve.

*** ICVLEW(NCVLVW)**
the element number of a check valve (initially entered as the user's number for the element, then changed to the code's number).

*** ICVSGN(M,NSGN)**
for M=1, ICVSGN is the user's number of the compressible volume of a steam generator inlet plenum; for M=2, it is the volume number of the outlet plenum.

*** IELPW(NPUMPW)**
element number of waterside pump (IELPMP).

*** IELVLW(NVLVW)**
element number of a valve (ordered by the code-generated valve number).

*** IEMPW(NPUMPW)**
type of waterside pump (IEMPMP).

*** IFBWCL(NCVW)**
flags whether a flow boundary condition is controlled by a table or by the control system, with IFBWCL = 0 if the boundary condition is controlled by a table, = 1 if the boundary condition is controlled by the control system.

*** IHTLW(NSEGW)**
the user's number of the volume containing the drain to which the segment is attached.

*** IHTUP(NSEGW)**
the user's number of the volume containing the de-superheating section to which the segment is attached.

*** IHTSEG(NSEGW)**
the user's number of the heater volume (if any) through which the segment passes.

ILEGW(NLEGS)
the number of compressible volumes in each leg of the water side.

*** ILRPW(NPUMPW)**
flag for locked rotor (ILRPMP).

*** IPMWCL(NPUMPW)**
control system flag for waterside pumps (IPMPCL).

*** IRVLVW(NRVLVW)**
the user's number for the element assigned to a check valve.

ISEGCV(NCVW,6)
segment numbers of segments attached to each compressible volume (maximum of 6 segments currently allowed).

*** ISGIN**
entries (1-10), user number of first segment in leg,

entries (11-20), code-generated number of first flow boundary condition pseudo-segment in leg,

entries (21-30), code-generated number of first steam generator inlet pseudo-segment in leg.

ISGNCV(-NCVW,6)
for each compressible volume, ISGNCV identifies the flow from each segment attached to the volume as flowing into or out of the volume at steady state. ISGNCV = 1 indicates flow into the volume, while YSGNCV = -1 indicates flow out of the volume.

*** ISGOUT**
entries (1-10), user number of last segment in leg,

entries (11-20), code-generated number of last flow boundary condition pseudo-segment in leg,

entries (21-30), code-generated number of last steam generator inlet pseudo-segment in leg.

*** ITYPW(NELEW)**
element type for each element (ITYPEL).

*** IVBWCL(NCVW)**
flags whether a volume boundary condition is controlled by a table or by the control system, with IVBWCL = 0 if the boundary condition is controlled by a table, = 1 if the boundary condition is controlled by the control system.

IVLELW(NVLVW)
code-generated element number of a valve (ordered by the code-generated valve number).

*** IVLWCL(NVLVW)**
control system flag for waterside valves (IVLVCL).

IVLWCL = 0 if the control system does not control the valve,

IVLWCL = 1 if the control system controls the valve driving function, and

IVLWCL = 2 if the control system controls the valve stem position directly.

*** JCVW(M,NSEGW)**
compressible volume numbers at each end of a segment (M=1 at the flow inlet, M=2 at the flow outlet) (JCVL).

*** JCVlFG(NSEGW)**
indicates where a segment attached to a heater volume is attached to the volume, with

JCVlFG = -1 if the segment is attached to the bottom of the volume,

= 0 if the segment is attached in between the top and the bottom of the volume,

= 1 if the segment is attached to the top of the volume.

*** JFSEW(NSEGW)**
first element in a segment (JFSELL).

*** JLSEW(NSEGW)**
last element in a segment.

*** JPRINT (17)**
an array of flags through which the user selects which parameters to include in the full balance-of-plant print.

LEGBCK(NLEGS)
the translator array from the user's numbering of the legs on the balance of-plant side to the code's internal numbering of the legs.

*** LEGORD(NLEGS)**
lists the order in which the legs into which the balance-of-plant is divided should be ordered in the output listing.

*** LMPDOT**
the number of steam generator timesteps averaged to compute the time derivative of pressure in the steam generator.

*** NBCCVF(NBCFLO)**
the number of the compressible volume to which the flow boundary condition pseudo-segment is attached (input as the user's c.v. number, then changed to the code's number).

*** NBCCVP(NBCPRS)**
the number of the compressible volume which serves as a boundary condition (input as the user's c.v. number, then changed to the code's number).

NBCFLO
number of flow boundary condition tables.

*** NBCINF(NBCFLO)**
table number for the time-dependent data for the flow boundary conditions.

*** NBCINP(NBCPRS)**
table number for the time-dependent data for the compressible volume boundary conditions.

NBCINT
number of interior volumes (volumes which are not boundary condition volumes).

*** NBCSEG(NBCFLO)**
code-generated pseudo-segment number for each flow boundary condition.

NBCPRS
number of volume boundary condition tables.

*** NBOREL(M,NELEW)**
neighboring element numbers for each element (M=1 for the upstream neighbor, M=2 for the downstream neighbor). NBOREL(1,I) = 0 for the first element in a segment and NBOREL(2,I) = -1 for the last element in a segment.

*** NCHVST(NCVLVW)**
flags the state or each check valve as follows:

= 1, valve is fully open and will begin to close if the pressure drop across the valve is less than the user-input value CHEPS1.

= 2, valve is fully open and will begin to close if the flow through the valve is less than CHEPSl.

= 3, valve is in the process of closing.

= 4, valve is fully closed but leaking slightly and will begin to open if the pressure drop across the valve becomes greater than the user-input value CHEPS2.

= 5, valve is fully closed but leaking slightly and will begin to open if the flow through the valve becomes greater than CHEPS2.

= 6, valve is in the process of opening.

*** NCVBCW**
identifies compressible volumes as boundary condition, steam generator plenum, etc., with NCVBCW

= 0 for a standard interior volume,

= 1 for a volume boundary condition volume,

= 2 for an inlet flow boundary condition volume,

= 3 for an outlet flow boundary condition volume,

= 4 for a steam generator inlet plenum,

= 5 for a steam generator outlet plenum,

= 6 for a heater volume,

= 7 for a turbine.

NCVIN(NLEGS)
user number of first compressible volume in loop.

NCVLBK(NCVLVW)
array which maps the user's number for a check valve to the code's number for that check valve.

NCVLTR(NCVLVW)
array which maps the code's number for a check valve to the user's number for the same check valve.

NCVLVW
number of check valves in the balance-of-plant loop.

NCVOUT(NLEGS)
user number of last compressible volume in loop.

NCVQ(NHTR)
code-generated compressible volume number of a heater (by code-generated heater number).

NCVW
number of compressible volumes (NCVT).

NELEW
number-of elements (NELEMT).

*** NELSGW(NELEW)**
user's number of the segment in which an element lies.

*** NELSUH**
the user's element number for a superheater.

*** NENTRF(NCVW)**
flag for the type of floating-point input data entered for a volume, with NENTRF

= 1 for single-phase volumes, pressure and temperature entered,

= 2 for single-phase volumes, pressure and enthalpyentered,

= 3 for two-phase volumes, pressure and quality entered,

= 4 for two-phase volumes, temperature and quality entered,

= 5 for two-phase heater volumes, pressure, two-phase level and ambient temperature entered,

= 6 for two-phase heater volumes, temperature, two-phase level, and ambient temperature entered.

*** NFLSEG(NCVW)**
flags the type of floating point data entered for an inflow boundary condition, with NFLSEG

= 0 if enthalpy is entered,

= 1 if temperature and pressure are entered for a subcooled liquid boundary condition,

= 2 if temperature and pressure are entered for a superheated steam boundary condition,

= 3 if quality and pressure are entered for a two-phase boundary condition,

= 4 if quality and temperature are entered for a two-phase boundary condition.

NHTR
number of heaters in the balance-of-plant.

NLEGS
number of legs (a leg is a section of the balance of plant for which all flows and volume pressures are solved

simultaneously. For example, the volumes and segments from the inlet to the steam generator might be one leg (a liquid leg), and those from the steam generator to the outlet might be another leg (a vapor leg).

*** NLGCVW(NCVW)**
the number of the leg of the loop to which a volume belongs.

NLVOL
number of liquid compressible volumes.

*** NODMAX(NSEGW)**
the maximum number of enthalpy transport nodes into which a segment may be divided.

*** NODSC**
the number of nodes in the evaporator subcooled zone.

*** NODSH**
the number of nodes in the evaporator superheated zone.

*** NODTP**
the number of nodes in the evaporator two-phase zone.

*** NODSHT**
the number of nodes in the superheater.

*** NOSGW(NSGN)**
user's number for the segment which is at the outlet of the vapor leg which is fed by the steam generator (used for saving plot data only).

NPUMPW
number of pumps in the balance-of-plant.

*** NPUTRN(NPUMPW)**
user's number of pump.

*** NQFLG(NCVW)**
user-assigned heater number for a compressible volume which is a heater.

NSEGCV(NCVW)
number of segments attached to each compressible volume.

NSEGT
the number of flow segments entered by the user (NSEGLT).

NSEGW
total number of segments, including pseudo-segments generated by flow boundary conditions and steam generator interfaces.

*** NSSIN(NSSEG)**
the compressible volume number at a supersegment inlet.

*** NSSOUT(NSSEG)**
the compressible volume number at a supersegment outlet.

*** NSUPSG(NCVW)**
the number of the supersegment in which a vapor volume is contained.

*** NTABVL(NBCPRS)**
flags the types of parameters entered in the floating point volume boundary condition table, with NTABVL

= 1 for pressure and enthalpy entered for a liquid volume

= 2 for pressure and temperature entered for a liquid volume,

= 3 for pressure and enthalpy entered for a vapor volume,

= 4 for pressure and temperature entered for a vapor volume,

= 5 for pressure and quality entered for a two-phase volume,

= 6 for temperature and quality entered for a two-phase volume.

*** NTPCVW(NCVW)**
compressible volume type, with NTPCVW

= 1 for a subcooled liquid volume,

= 2 for a superheated vapor volume,

= 3 for a two-phase volume,

= 4 for a pseudo-volume at the liquid/two-phase interface in an evaporator.

NTPELW(NELEW)
state of an element, with NTPELW

= 1 for a subcooled liquid element,

= 2 for a superheated vapor element,

= 3 for a two-phase element.

*** NTRNPT**
flags whether or not enthalpy transport is used in the vapor leg, with NTRNPT

= 0 if enthalpy transport is used,

= 1 if enthalpy transport is not used.

NVLBCK(NVLVW)
array which takes the number assigned to a valve by the user and gives the number assigned to the valve by the code.

NVLTRN(NVLVW)
array which takes the code-generated number assigned to a valve and gives the number assigned to the valve by the user.

NVLVW
number of valves in the balance-of-plant (NVALVE).

*** OMEGAR**
turbine rotor angular velocity.

*** ORIFLW(NCVW)**
the elevation of the drain orifice in a heater which contains a drain.

*** ORIFUP(NCVW)**
the elevation of the desuperheating region orifice in a heater which contains a desuperheating region.

*** PCVI**
the initial upstream pressure for a relief valve.

*** PCVO**
the initial downstream pressure for a relief valve.

*** PELEW(NELEW)**
pressure at an element outlet.

*** PMPFWR(NPUMPW)**
rated pump flow (PMPFLR).

PMPHDW
coefficients in centrifugal pump option 2 (PMPHD).

*** PMPSWR(NPUMPW)**
rated pump speed (PMPSPR).

PMPTQW
torque coefficients in cent. pump option 2 (PMPTQ).

*** PMWEFR(NPLPMPW)**
pump efficiency (PMPEFR).

*** PMWINR(NPUMPW)**
moment of inertia, pump and motor (PMPINR).

PMWTQR(NPUMPW)
steady state pump torque (PMPTQR).

*** PRESW4(NCVW)**
pressure in each compressible volume at the end of the current PRIMAR time subinterval (PRESL4). Pressure at the beginning of the subinterval is PRESW3, and the pressure at the end of the PRYMAR timestep is PRESW2.

PSPDW2(NPUMPW)
pump speed at start of PRIMAR timestep (PSPED2).

PSPDW3(NPUMPW)
pump speed at start of timestep (PSPED3).

PSPDW4(NPUMPW)
pump speed at end of timestep (PSPED4).

*** QRATIO(NCVW)**
the percentage of incoming energy to a heater lost due to imperfect insulation.

*** ROUGHW(NELEW)**
the roughness of an element wall (ROUGHL).

*** RROTOR(NCVW)**
the radius of a turbine rotor.

*** RVA(NRVLVW)**
the fractional valve area to which a relief valve opens when the set pressure drop is reached.

*** RVFRAC(NRVLVW)**
the fractional relief valve opening area.

SEGLW(NSEGW)
length of a segment.

*** SHHTCC(NCVW)**
the shell side condensation coefficient for a heater.

*** TABSEG(10,3,NBCFLO)**
table for flow boundary condition input data. TABSEG(x,1,y) contains time, TABSEG(x,2,y) contains absolute flows, and TABSEG(x,3,y) contains enthalpies.

*** TABVOL(10,4,NBCPRS)**
table for compressible volume boundary condition input data. TABVOL(x,1,y) contains time, TABVOL(x,2,y) contains pressures, TABVOL(x,3,y) contains enthalpies, and TABVOL(x,4,y).contains qualities.

*** TAMBNT(NCVW)**
the ambient temperature for a heater volume.

*** TBCP(NELEW)**
the specific heat of the tube in an element representing a heater tube bundle.

*** TBKPMO(NELEW)**
the thermal conductivity of the tube in an element representing a heater tube bundle.

*** TBLNLW(NELEW)**
the length of the section of the element within the drain for an element representing a tube bundle in a drain cooler or desuperheater/drain cooler.

*** TBLNUP(NELEW)**
the length of the section of the element within the desuperheating section for an element representing a tube bundle in a desuperheating heater or a desuperheater/drain cooler.

*** TBNDLW(NELEW)**
the number of nodes for the section of the element within the drain for an element representing a tube bundle in a drain cooler or desuperheater/drain cooler.

*** TBNDUP(NELEW)**
the number of nodes for the section of the element within the superheating section for an element representing a tube bundle in a desuperheating heater or desuperheater/drain cooler.

*** TBNMBR(NELEW)**
the total number of tubes in a heater tube bundle.

*** TBNODE(NELEW)**
the number of nodes for the heat transfer calculation in an element representing a heater tube bundle.

*** TBPODS**
the superheater bundle pitch-to-diameter ratio.

*** TBRHO(NELEW)**
the tube material density in an element representing a heater tube bundle

*** TBTHIK(NELEW)**
the tube thickness in an element representing a heater tube bundle.

*** TCVW(NCVW)**
compressible volume temperature (TLQCV2).

*** TEMPLW(NCVW)**
the temperature of the drain in a heater containing a drain.

*** TEMPUP(NCVW)**
the temperature in the desuperheating region in a heater containing a desuperheating region.

*** TIMERV(NRVLVW)**
the relief valve delay time for opening or closing.

*** TPFACE(NCVW)**
the two-phase level in a volume in which liquid and vapor are separated.

TQMBW3(NPUMPW)
motor torque at start of timestep (TQMB3).

TQMBW4(NPUMPW)
motor torque at end of timestep (TQMB4).

TQPBW3(NPUMPW)
pump torque at start of timestep (TQPB3).

TQWSAV(NPUMPW)
torque from PUMPFL (TQBSAV).

*** TRGRMI**
turbine/generator rotor moment of inertia.

*** TRKLSW(NPUMPW)**
windage (TRKLSC).

TRQMSW(NPUMPW)
initial steady state speed (TRQMSS).

*** TSECHT(NHTR)**
the temperature of the secondary fluid in a heater.

*** TUBNOS**
the number of superheater tubes.

VCALBW(NVLVW)
calibration constant for a standard valve.

*** VCONSW(NVLVW)**
the proportionality constant between the stem position and the valve characteristic for a standard valve.

*** VDAMPW(NVLVW)**
damping coefficient for the valve stem position equation.

VDRIVW(NVLVW)
driving function for the valve stem position equation.

*** VLVMSW(NVLVW)**
valve mass.

*** VPHINW(NVLVW)**
valve characteristic at the current PRIMAR subinterval.

VPHIW(10,NVLVW)
valve characteristic curve for a standard valve.

*** VPOSW(10,NVLVW)**
valve stem position for points in VPHIW.

*** VSPRGW(NVLVW)**
spring constant for the valve stem position equation.

*** VSTEMW(NVLVW)**
valve stem position.

VSTMWl(NVLVW)
valve stem position from the previous timestep.

*** VTABDW(10,NVLVW)**
table of driving function vs. time for a standard valve (this array is used to vary driving function with time if the control system is not used to control the valve).

*** VTIMW(10,NVLVW)**
values of time for VTABDW.

*** VOLCVW(NCVW)**
volume of each compressible volume (VOLLGC).

*** VOLLW(NCVW)**
the volume of the drain in a heater containing a drain.

*** VOLUP(NCVW)**
the volume of the desuperheating region in a heater containin desuperheating region.

*** WMOTTK(20,NPUMPW)**
motor torque table and times (AMOTTK).

*** XCVW(NCVW)**
compressible volume quality.

*** XKTUBE**
the evaporator tube thermal conductivity.

*** XLENLW(NCVW)**
the length of the drain in a heater containing a drain.

*** XLENUP(NCVW)**
the length of the desuperheating region in a heater containing a desuperheating region.

*** XLENW(NELEW)**
length of an element (XLENEL).

*** XRXFR(NSEG)**
nozzle reaction fraction.

*** ZCVW(NCVW)**
compressible volume midpoint elevation (ZCVL).

*** ZINW(NSEGW)**
elevation of the segment inlet (ZINL).

*** ZLOWST(NELEW)**
the lowest elevation of the element within the heater for an element representing a heater tube bundle.

*** ZONLE(3)**
the zone lengths in the evaporator. ZONLE(1) is the subcooled zone length, and ZONLE(3) is the superheated zone length; these are both input, with ZONLE(2) (the two-phase zone length) calculated from EL, ZONLE(1), and ZONLE(3) (ELEV).

*** ZOUTLW(NELEW)**
elevation of the element outlet (ZOUTEL).

7.7.2 Steam Generator Water-Side Heat Transfer Correlations

Subcooled Water

The Dittus-Boelter correlation *[7-4]* is used.

$$N = 0.023 Re^{0.8} Pr^{0.4} = \frac{h_w D_H}{k}$$

Bulk liquid properties are used to calculate Re, Pr and k; h_w is the heat transfer coefficient between the wall surface and the bulk water.

Nucleate Boiling Water

A correlation developed by Thom, et al. *[7-5]* is used.

$$h_w = 3.1968 \left(e^{P/8.65\times10^6} \right) \frac{1}{0.072} (q)^{0.5}$$

h_w is the heat transfer coefficient between the wall surface and the bulk water; $q = H_T (T_m - T_{sat})$ where H_T is defined as in Eq. (7.3-66); T_m is the average wall temperature; P is the steam generator pressure.

Film Boiling Water

A correlation of A. A. Bishop et al. *[7-6]* is the following

$$N = 0.0193 Re^{0.8} Pr^{1.23} \left[x + (1-x) \frac{\rho_g}{\rho_f} \right]^{0.68} \left(\frac{\rho_g}{\rho_f} \right)^{0.068} = \frac{h_w D_H}{k}$$

A modification of the original formulation is used. The original formulation specified that properties appropriate for the wall film temperature be used to calculate Re and Pr. All temperature-dependent properties are calculated with T_{sat}. The wall film temperature is only crudely approximated and the relevant properties are very insensitive to temperature when above T_{sat}. h_w is the heat transfer coefficient between the wall surface and the bulk water and x is the local nodal quality. The mass flux used in the Reynold's number is the local value.

Superheated Steam

A correlation developed by A. A. Bishop *[7-7]* is used.

$$N = 0.0073 Re^{0.886} Pr^{0.61} = \frac{h_w D_H}{k}$$

Although the original correlation specified that the film temperature be used to evaluate the relevant properties, the bulk temperature is instead used since these properties are very insensitive to temperature above T_{sat}. h_w is the heat transfer coefficient between the wall surface and the bulk steam.

Liquid Sodium Heat Transfer

A variation of the Maresca-Dwyer correlation is used *[7-8]*. The heat transfer coefficient between the outside wall and the bulk sodium is the following.

$$H_{Na} = \frac{Nc_p\mu}{PrD_H}$$

where c_p, μ and D_H are the specific heat, viscosity and hydraulic diameter respectively. The Prandtl number is computed according to the following,

$$Pr = 0.00212 + 2.329/\left(1.8T - 410.92\right)$$

Nusselt numbers are computed for both turbulent flow and molecular conduction. The greater of the two is used. For turbulent flow,

$$Nu = 6.66 + 3.126POD + 1.184POD^2 + 0.0155\left(PrReS\right)^{0.86}$$

where POD is the tube pitch-to-diameter ratio. S is calculated according to the following,

$$S = 1.0 - 1.82\ /\ (PrE)$$

and E is given by,

$$E = 0.000175Re^{1.32}\ /\ POD^{1.5}$$

The Nusselt number for molecular conduction is given by the following,

$$Nu = 6.4353 + 3.97POD + 1.025POD^2 - 29494/(Re + 20363)$$

7.7.3 Two-Phase Interface Solution Scheme for Heater Cylinders Lying on the Side

The solution scheme described below is used to solve for the two-phase interface, for heater cylinders lying on the side, form the transcendental equations given in Eq. (7.4-8) and Eq. (7.4-9). Since the problem is symmetric with respect to the center point of the cylinder, as is obvious by looking at Figure 7.4.3, only Eq. (7.4-8) is used to demonstrate the solution scheme.

By defining the angle between the vertical section L and the radius r_s as β in Figure 7.4.3, Eq. (7.4-8) can be rewritten as,

$$\alpha_s = \left[\beta - \frac{\sin\left(2\beta\right)}{2}\right]/\pi \tag{7.7-1}$$

for $0 \leq \alpha_s \leq 1/2$ and $0 \leq \beta \leq \pi/2$. The height of the vapor region A_g corresponding to α_s in Figure 7.4.3 is $r_s - L$, which can be normalized to the cylinder radius r_s as

$$\omega \equiv \frac{r_s - L}{r_s} = 1 - \cos\beta \tag{7.7-2}$$

The aim now is to obtain an expression for β as a function of α_s, in order to avoid using an iterative solution to find β. The most straight-forward way would be to generate a polynomial expression in α_s for β. However, a study of the curvature of the β vs. α_s curve indicates that the slope is very steep at β close to zero, changes dramatically as β increases, and levels off as β approaches $\pi/2$. A single polynomial of high order to approximate

the curve is difficult to obtain without unacceptable errors in some part of the curve, and the computation of the polynomial may be time-consuming. Thus, an alternative method is used and is described as follows.

The range of α_s on the β vs. α_s curve is divided into three regions based on the slopes along the curve, i.e., $0 \leq \alpha_s < \alpha_1$, $\alpha_1 \leq \alpha_s < \alpha_2$, and $\alpha_2 \leq \alpha_s \leq 1/2$, where α_1 and α_2 are chosen to be at the suitable values 0.015 and 0.225, respectively. Also, the polynomials of β for each region are given the forms,

$$\beta = C_1 \left(1.5\alpha_s\right)^{1/3}, 0 \leq \alpha_s < \alpha_1 \tag{7.7-3}$$

$$\beta = \frac{\beta_1 + C_2 X_1 + C_3 X_1^2}{1 + C_4 X_1 + C_5 X_1^2}, \alpha_1 \leq \alpha_s < \alpha_2 \tag{7.7-4}$$

$$\beta = \frac{\beta_2 + C_6 X_2 + C_7 X_2^2}{1 + C_8 X_2 + C_9 X_2^2}, \alpha_2 \leq \alpha_2 < 1/2 \tag{7.7-5}$$

where β_1 and β_2 are values of β corresponding to α_1 and α_2, respectively, X_1 and X_2 are defined as

$$X_1 = (\alpha_2 - \alpha_1)$$

and

$$X_2 = (\alpha_s - \alpha_2)$$

and the coefficients C_1 through C_9 are determined using a least squares fit separately on each region. The coefficient values are as follows:

- $C_1 = 1.00377$
- $C_2 = 1.41595 \times 10^1$
- $C_3 = 6.98089 \times 10^1$
- $C_4 = 2.76847 \times 10^1$
- $C_5 = 4.47906 \times 10^1$
- $C_6 = 2.37819$
- $C_7 = 6.38891 \times 10^{-1}$
- $C_8 = 1.72658$
- $C_9 = 4.94197 \times 10^2$

More significant digits for C_1 through C_9 are used in the coding. The polynomials Eq. (7.7-3) through Eq. (7.7-5) are chosen such that continuity conditions are satisfied at α_s equal to α_1 and α_2.

Once β is calculated from one of Eq. (7.7-3), Eq. (7.7-4), and Eq. (7.7-5) for a given α_s, the two-phase interface can be computed as

$$TP = (1 - \omega)\, r + CV,$$

where ω is given in Eq. (7.7-2).

The maximum error, defined as the difference between ω, as calculated from Eq. (7.7-2) with β obtained by Eq. (7.7-3), Eq. (7.7-4), or Eq. (7.7-5), and the actual ω, is within $\pm 4.53 \times 10^{-4}$. If more accuracy is needed, a better value for ω can be obtained by introducing the Newton iteration method and using the calculated ω With one iteration, the maximum error could be reduced to $\pm 2.67 \times 10^{-7}$, and with two iterations, to $\pm 9.17 \times 10^{-13}$.

7.7.4 Dictionary of Steam Generator Model Variables

Variables in COMMON

Note: Variables in block SGEN1, SGEN2, and SGEN3 apply to the once-through steam generator or to the evaporator in the recirculation type steam generator. Variables in blocks SGENS1, SGENS2, and SGENS3 apply to the superheater only.

Name	Block	Units	Explanation
ARM	SGEN2	m^2	Cross-sectional area of tube wall
ARMS	SGENS2	m^2	Cross-sectional area of tube wall
ARNA	SGEN2	m^2	Flow area of sodium
ARNAS	SGENS2	m^2	Flow area of sodium
ARW	SGEN2	m^2	Flow area of water
ARWS	SGENS2	m^2	Flow area of water
AWB(100)	SGEN1	-	Void fraction at each node in boiling zone at beginning of time step
AWE(100)	SGEN1	-	Void fraction at each node in boiling zone at end of time step
CNAFRS	SGENS2	-	Constant used in sodium heat transfer coefficient calculation; equal to $6.66 + (1.184TBPODS + 3.126)\,TBPODS$
CNAFR1	SGEN2	-	Constant used in sodium heat transfer coefficient calculation; equal to $6.66 + (1.184TUBPOD + 3.126)\,TUBPOD$
COILD	SGEN2	m	Average diameter of coil in helical coil in the helical coil geometry option for evaporator/steam generator model
COILDS	SGENS2	m	Average diameter of the coil for the helical coil geometry option in the superheater model
DDW	SGEN2	m	Hydraulic diameter on water side
DDWS	SGENS2	m	Hydraulic diameter on water side
DELP24	SGEN2	-	Fraction of total pressure drop in subcooled zone
DEWI	SGEN2	m	Booster tube outer diameter on water side
DEWIS	SGENS2	m	Booster tube outer diameter on water side
DEWO	SGEN2	m	Tube wall inner diameter
DEWOS	SGENS2	m	Tube wall inner diameter
DOUT	SGEN2	m	Tube wall outer diameter
DOUTS	SGENS2	m	Tube wall outer diameter
DHNA	SGEN2	m	Hydraulic diameter on sodium side
DHNAS	SGENS2	m	Hydraulic diameter on sodium side
DZONE(2)	SGEN2	m/s	Velocity of subcooled and superheat zone boundaries
FACT1	SGEN2	-	Not currently used
FACT1S	SGENS2	-	Not currently used
FACT2	SGEN2	-	$1.04 * 10^4 * TUBPOD^{1.5}$; used in sodium heat transfer coefficient calculation
FACT2S	SGENS2	-	$1.04 * 10^4 * TBPODS^{1.5}$; used in sodium heat transfer coefficient calculation
FACT3	SGEN2	m^{-1}	$\pi * DEWO/ARW$

continues on next page

Table 7.7.1 – continued from previous page

Name	Block	Units	Explanation
FACT3S	SGENS2	m^{-1}	$\pi * DEWOS/ARWS$
FACT4	SGEN2	m^{-1}	$\pi * DOUT/ARNA$
FACT4S	SGENS2	m^{-1}	$\pi * DOUTS/ARNAS$
FACT5	SGEN2	-	$ARNA/ARM$
FACT5S	SGENS2	-	$ARNAS/ARMS$
FACT6	SGEN2	-	ARW/ARM
FACT6S	SGENS2	-	$ARWS/ARMS$
FACT7	SGEN2	K	Constant used in viscosity function
FACT8	SGEN2	kg/m^3	Constant used in viscosity function
FACT9	SGEN2	$J/m^3 - K$	Density x specific heat for tube wall
FACT10	SGEN2	-	$0.023 * DDW^{-0.2}$; used in subcooled heat transfer coefficient
FACT11	SGEN2	-	$0.0193 * DDW^{-0.2}$; used in film boiling heat transfer coefficient
FACT12	SGEN2	-	$0.0073 * DDW^{-0.114}$; used in superheat zone heat transfer coefficient
FACTSP	SGENS2	-	$0.0073 * DDWS^{-0.114}$; used in superheater heat transfer coefficient
FOULR(4)	SGEN2	$m^2 - K/w$	Tube wall heat resistance on the water side plus any fouling heat resistance on water side for each heat transfer regime
FOULRI(4)	SGEN2	$m^2 - K/w$	Fouling heat resistances on the water side for each heat transfer regime
FOULRS	SGENS2	$m^2 - K/w$	Tube wall heat resistance on the water side plus any fouling heat resistance on water side
FOULSI	SGENS2	$m^2 - K/w$	Fouling heat resistance on the water side
FRIC1(4)	SGEN2	-	Normalizing friction factor in Eq. (7.3-56) and Eq. (7.3-94) for each heat transfer regime
FRIC1S	SGENS2	-	Normalizing fraction factor in superheater
GNA	SGEN2	$kg/m^2 - s$	Sodium side mass flow
GNAS	SGENS2	$kg/m^2 - s$	Sodium side mass flow
GWB(100)	SGEN1	$kg/m^2 - s$	Water side mass flow at each node at beginning of time step
GWE(100)	SGEN1	$kg/m^2 - s$	Water side mass flow at each node at end of time step
GWS	SGENS2	$kg/m^2 - s$	Sodium side mass flow
HD	SGEN2	(J/kg)/(BTU/lb)	Conversion factor for enthalpies since functions are in (BTU/lb)
HDNB	SGEN2	-	Fraction of cell where DNB point lies which is in the nucleate boiling regime
HFG	SGEN2	J/kg	h_{fg}
HFSAT	SGEN2	J/kg	h_f
HFSATP	SGEN2	BTU/lb	h_f/HD
HGSAT	SGEN2	J/kg	h_g
HGSATP	SGEN2	BTU/lb	h_g/HD
HN(520)	SGEN2	s	Array which stores all potential time steps from which is selected the minimum
HSTEP	SGEN2	s	Primary loop time step

continues on next page

Table 7.7.1 – continued from previous page

Name	Block	Units	Explanation
HTF(4)	SGEN2	-	Calibration factors for heat transfer coefficients for each regime
HTFI(4)	SGEN2	-	Calibration factors for heat transfer coefficients for each regime
HTFS	SGENS2	-	Calibration factor for heat transfer coefficient in superheater
HTW(100)	SGEN2	$w/m^2 - K$	Heat transfer coefficient between the tube wall surface and the bulk water by cell center
HTWS	SGENS2	$w/m^2 - K$	Heat transfer coefficient between the tube wall surface and the bulk water by cell center in superheater
HUNIT	SGEN2	s	Current steam generator time step
HUNITN	SGEN2	s	Newly selected steam generator time step for next step
HUNITS	SGENS2	s	Current superheater time step
HWB(100)	SGEN1	J/kg	Enthalpy by node at beginning of step
HWBS(100)	SGENS1	J/kg	Enthalpy by node at beginning of step
HWE(100)	SGEN1	J/kg	Enthalpy by node at the end of step
HWES(100)	SGENS1	J/kg	Enthalpy by node at the end of step
H1MIN	SGENS2	$w/m^2 - K$	Minimum value allowed for HTW in subcooled zone
H2MIN	SGEN2	$w/m^2 - K$	Minimum value allowed for HTW in nucleate boiling zone
H3MIN	SGEN2	$w/m^2 - K$	Minimum value allowed for HTW in film boiling zone
H4MIN	SGEN2	$w/m^2 - K$	Minimum value allowed for HTW in superheated zone or for :math:HTWS` in superheater
IDNB	SGEN3	-	Cell number when DNB point occurs
IDNBL	SGEN3	-	Value of $IDNB$ during previous time step
ISTEPW	SGEN3	-	Number of current primary loop time step
LAR	SGEN3	-	Array size limit for nodal arrays
LIM	SGEN3	-	Length of $SGEN1COMMON$ block
NCOUNT	SGEN3	-	Number of steam generator time substeps within primary loop step
NODSC	SGEN3	-	Number of cells within subcooled zone
NODSCO	SGEN3	-	Initial value of $NODSC$
NODSC1	SGEN3	-	Node number of subcooled/boiling boundary; $NODSC + 1$
NODSC2	SGEN3	-	$NODSC1 + 1$
NODSH	SGEN3	-	Number of cells within superheated zone in steam generator or evaporator
NODSHO	SGEN3	-	Initial value of $NODSH$
NODSHT	SGENS3	-	Number of cells within superheater
NODSH0	SGEN3	-	Total number of cells in steam generator or evaporator; $NODSC + NODTP + NODSH$
NODSH1	SGEN3	-	$NODSHT + 1$; or total number of nodes in superheater
NODT	SGEN3	-	$NODSH0 + 1$; or total number of nodes in steam generator or evaporator
NODTP	SGEN3	-	Number of cells within boiling zone

continues on next page

Table 7.7.1 – continued from previous page

Name	Block	Units	Explanation
NODTPO	SGEN3	-	Initial value of $NODTP$
NODTP0	SGEN3	-	$NODSC + NODTP$
NODTP1	SGEN3	-	Node number of boiling/superheat boundary; $NODSC + NODTP + 1$
NODTP2	SGEN3	-	$NODTP1 + 1$
ON	SGEN2	-	1.0
PD	SGEN2	Pa/PSI	Conversion factor for pressures since functions are in PSI
PDOT	SGEN2	Pa/s	Time derivative of steam generator average pressure
PI	SGEN2	-	π
PICHL	SGEN2	m	Longitudinal pitch of the helical tubes in the helical coil geometry option for the evaporator/steam generator model
PICHLS	SGENS2	m	Longitudinal pitch of the helical tube in the helical coil geometry option for the superheater model
PICHT	SGEN2	m	Transverse pitch of the helical tubes in the helical coil geometry option for the evaporator/steam generator model
PICHTS	SGENS2	m	Transverse pitch of the helical tube in the helical coil geometry option for the superheater model
PSW	SGEN2	-	ρ_g/ρ_f
PWAVEP	SGEN2	PSI	$PWAVES/PD$
PWAVES	SGEN2	Pa	Steam generator pressure
PWAVSP	SGENS2	Pa	Pressure in superheater
PWVSPP	SGENS2	PSI	$PWAVSP/PD$
P25	SGEN2	-	0.25
P5	SGEN2	-	0.5
QMT(100)	SGEN2	w/m^3	Volumetric heat source for each cell in the tube wall
QMTS(100)	SGENS2	w/m^3	Volumetric heat source for each cell in the tube wall
QST(100)	SGEN2	w/m^3	Volumetric heat source for each cell in the sodium side
QSTS(100)	SGENS2	w/m^3	Volumetric heat source for each cell on the sodium side
QWB(100)	SGEN1	w/m^3	Volumetric heat source for each cell on the water side at the beginning of step
QWBS(100)	SGENS1	w/m^3	Volumetric heat source for each cell on the water side at the beginning of step
QWE(100)	SGEN1	$w/m^2 - K$	Total heat transfer coefficient from tube wall center to bulk water including possible fouling for each cell center
QWES(100)	SGENS1	$w/m^2 - K$	Total heat transfer coefficient from tube wall center to bulk water including possible fouling for each cell center
QWT(100)	SGEN2	w/m^3	Volumetric heat source for each cell on the water side at the end of step
QWTS(100)	SGENS2	w/m^3	Volumetric heat source for each cell on the water side at the end of step
RMDEWO	SGEN2	$m^2 - K/w$	Tube wall heat resistance on the water side
RMDEWS	SGENS2	$m^2 - K/w$	Tube wall heat resistance on the water side

continues on next page

Table 7.7.1 – continued from previous page

Name	Block	Units	Explanation
RMDNAA	SGEN2	$m^2 - K/w$	Tube wall heat resistance on the sodium side
RMDNAS	SGENS2	$m^2 - K/w$	Tube wall heat resistance on the sodium side
ROB(100)	SGEN1	kg/m^3	Water density at each node at beginning of step
ROBS(100)	SGENS1	kg/m^3	Water density at each node at beginning of step
ROCPTB	SGEN2	-	Reserved
ROE(100)	SGEN1	kg/m^3	Water density at each node at end of step
ROES(100)	SGENS1	kg/m^3	Water density at each node at end of step
ROFG	SGEN2	kg/m^3	$\rho_g - \rho_f$
ROFSAT	SGEN2	kg/m^3	ρ_f
ROGSAT	SGEN2	kg/m^3	ρ_g
ROHFG	SGEN2	J/m^3	$\rho_g h_g - \rho_f h_f$
ROZ1	SGEN2	kg/m^3	Average water density in subcooled zone
TBPODS	SGENS2	-	Tube pitch-to-diameter ratio
TIMCUR	SGEN2	s	Time at end of current steam generator time step
TIMEIN	SGEN2	s	Time at beginning of current primary loop time step
TIMENP	SGEN2	s	Time at end of current primary loop time step
TLIM	SGEN2	-	Largest fractional change in selected parameters for new time step selection
TMB(100)	SGEN1	K	Temperature at each cell center of tube wall at beginning of step
TMBS(100)	SGENS1	K	Temperature at each cell center of tube wall at beginning of step
TME(100)	SGEN1	K	Temperature at each cell center of tube wall at end of step
TMES(100)	SGENS1	K	Temperature at each cell center of tube wall at end of step
TO	SGEN2	-	2.0
TSB(100)	SGEN1	K	Temperature of sodium at each node at beginning of step
TSBS(100)	SGENS1	K	Temperature of sodium at each node at beginning of step
TSC(100)	SGEN2	K	Temperature of sodium at each node at beginning of step
TSCS(100)	SGENS2	K	Temperature of sodium at each cell center at beginning of step
TSE(100)	SGENS2	K	Temperature of sodium at each node at end of step
TSES(100)	SGENS1	K	Temperature of sodium at each node at end of step
TUBNO	SGEN2	-	Number of tubes in steam generator
TUBNOS	SGENS2	-	Number of tubes in superheater
TUBPOD	SGEN2	-	Tube pitch-to-diameter ratio
TWB(100)	SGEN1	K	Temperature of water at each node at beginning of step
TWBS(100)	SGENS1	K	Temperature of water at each node at beginning of step
TWC(100)	SGENS1	K	Temperature of water at each cell center at beginning of step
TWCS(100)	SGENSG	K	Temperature of water at each cell center at beginning of step

continues on next page

Table 7.7.1 – continued from previous page

Name	Block	Units	Explanation
TWE(100)	SGEN1	K	Temperature of water at each node at end of step
TWES(100)	SGENS1	K	Temperature of water at each node at end of step
TWSAT	SGEN2	K	Water saturation temperature
UWZ1	SGEN2	$kg/m-s$	Average viscosity in the subcooled zone
VRISE	SGEN2	-	Vertical rise per length of helical tube in the helical coil geometry option for the evaporator/steam generator model
VRISES	SGENS2	-	Vertical rise per length of helical tube in the helical coil geometry option in the superheater model
XKTUBE	SGEN2	$W/m-K$	Conductivity of tube wall
XWB(100)	SGEN1	-	Quality at each node in boiling zone at beginning of step
XWE(100)	SGEN1	-	Quality of each node in boiling zone at end of step
ZMAX	SGEN2	m	Zone length threshold above which the number of nodes is restored to the initial value when the previous number of nodes is one
ZMIN	SGEN2	m	Zone length threshold below which the number of nodes is reduced to one
ZO	SGEN2	-	0.0
ZONLB(3)	SGEN2	m	Lengths of each zone at beginning of step
ZONLE(3)	SGEN2	m	Lengths of each zone at end of step
ZSG	SGEN2	m	Length of steam generator or of evaporator
ZSUP	SGEN2	m	Length of superheater

Selected Variables not in COMMON

Name	Routine	Units	Explanation
AWZ2	SGUNIT INIT	-	Average void fraction in nucleate boiling zone
AWZ3	SGUNIT INIT	-	Average void fraction in film boiling zone
DELP1	SGUNIT INIT	Pa	Pressure drop across subcooled zone
DELP2	SGUNIT INIT	Pa	Pressure drop across nucleate boiling zone
DELP3	SGUNIT INIT	Pa	Pressure drop across film boiling zone
DELP4	SGUNIT INIT	Pa	Pressure drop across superheated zone
DHF	SGUNIT	$J/kg-Pa$	Derivative of h_f with respect to pressure
DHG	SGUNIT	$J/kg-Pa$	Derivative of h_g with respect to pressure
DRODH	SGUNIT	kg^2/m^3-J	Derivative of respect to enthalpy in superheated zone
DRODP	SGUNIT	kg/m^3-Pa	Derivative of $\rho\left(h,P\right)$ with respect to pressure in superheated zone
DROF	SGUNIT	kg/m^3-Pa	Derivative of ρ_f with respect to pressure
DROG	SGUNIT	kg/m^3-Pa	Derivative of ρ_g with respect to pressure

continues on next page

Table 7.7.2 – continued from previous page

Name	Routine	Units	Explanation
DTSG(100)	TSBOP	s	Array to store time steps over $LMPDOT$ steam generator time steps in order to calculate.
GDOT	SGUNIT	$kg/m^2 - s$	in pressure drop calculation
GWO	INIT	$kg/m^2 - s$	Steady state mass flow
GWZ	SGUNIT INIT	$k/m^2 - s$	End of time step regional average mass flow for pressure drop calculation
GWZ0	SGUNIT	$k/m^2 - s$	Beginning of time step regional average mass flow for pressure drop calculation
HTAV	INIT	$W/m^2 - K$	Average heat transfer coefficient at tube wall surface on water side in one-node approximation or region
HTAVT	INIT	$W/m^2 - K$	Total water side average heat transfer coefficient in one-node approximation of region
HWAV	INIT	J/kg	Average water enthalpy in one-node approximation of region
HWIN	INIT	J/kg	Steady state inlet water enthalpy
HWOUT	INIT	J/kg	Steady state outlet water enthalpy
IGO	INIT	-	Indicator which is set when the DNB node is found in the boiling zone so that a switch is made from the nucleate to the film boiling regime
INITER	INIT	-	Counter on the number of iterations in the search on the calibration factor in the film boiling calculation for each iteration on the nucleate boiling regime
IOPT1	INIT	-	Indicator which shows whether length $(= 2)$ or calibration factor $(= 1)$ is to be searched on for subcooled zone
IOPT2	INIT	-	Indicator which shows whether length $(= 2)$ or calibration factor $(= 1)$ is to be searched on for superheated zone
IOPT3	INIT	-	Indicator which shows how many heat transfer regimes there are in the steady state calculation
IPASS	SGUNIT	-	Indicator which stops iterative search on Z_{TP} in boiling zone when the zone reaches the top of the steam generator or when Z_{TP} changes more than a maximum amount allowed
ITER	SGUNIT INIT	-	Iteration counter either on boiling zone length searches in $SGUNIT$ or searches in all three zones in $INIT$
LMPDOT	TSBOP	-	Number of time steps and pressures stored in $DTSG$ and $PTSG$ arrays for calculation
PTSG(100)	TSBOP	Pa	Array to store pressures over $LMPDOT$ steam generator time steps in order to calculate
ROAV	INIT	kg/m^3	Average water density in one-node approximation of region
ROEDNB	SGUNIT INIT	kg/m^3	Water density at the DNB point used in pressure drop calculation
ROZ2	SGUNIT INIT	kg/m^3	Average density in nucleate boiling zone for pressure drop calculation
ROZ3	SGUNIT INIT	kg/m^3	Average density in film boiling zone for pressure drop calculation

continues on next page

Table 7.7.2 – continued from previous page

Name	Routine	Units	Explanation
ROZ4	SGUNIT INIT	kg/m^3	Average density in superheated zone for pressure drop calculation
R32	SGUNIT INIT	-	Thom friction factor in nucleate boiling zone
R33	SGUNIT INIT	-	Thom friction factor in film boiling zone
TIMDIF	TSBOP	s	Time difference between beginning of primary loop time step and the end of current steam generator time step
TMAV	INIT	K	Average tube wall temperature in one-node approximation to region
TNAINB	TSBOP	K	Inlet sodium temperature at the beginning of primary loop time step
TNAINE	INIT	K	Inlet sodium temperature at the end of primary loop time step
TSAV	INIT	K	Average sodium temperature in one-node approximation to region
TSHF	INIT	K	Sodium temperature at the point of h_f on the water side at steady state
TSHG	INIT	K	Sodium temperature at the point of h_f on the water side at steady state
TSIN	INIT	K	Steady state sodium inlet temperature
TSOUT	INIT	K	Steady state sodium outlet temperature
TWAV	SGUNIT INIT	K	Average water temperature in one-node approximation to region
UZ2	SGUNIT INIT	$kg/m-s$	Average viscosity in nucleate boiling zone for pressure drop calculation
UZ3	SGUNIT INIT	$kg/m-s$	Average viscosity in film boiling zone for pressure drop calculation
UZ4	TSBOP	$kg/m-s$	Average viscosity in superheated zone for pressure drop calculation
WNAINB	TSBOP	kg/s	Sodium flow rate at beginning of primary loop time step
WNAINE	SGUNIT	kg/s	Sodium flow rate at end of primary loop time step
ZITER	SGUNIT INIT	m	Current value of Z_{TP} during search on region length in boiling zone calculation
ZONLE2	SGUNIT INIT	m	Length of nucleate boiling zone used in pressure drop calculation
ZONLE3		m	Length of film boiling zone used in pressure drop calculation

7.7.5 Material Properties Data

This Appendix documents material properties correlations employed throughout the balance-of-plant network model, the steam generator model, and the component models for thermal and physical properties data. These data are used in heat transfer and fluid dynamics calculations.

On the sodium side of the steam generator, the correlations used for liquid sodium density, liquid sodium specific heat, and liquid sodium viscosity are documented in Section 12.12.

On the water side of the steam generator and throughout the balance-of-plant models, the dynamic viscosity of steam and water is calculated from *[7-13]*:

$$\mu = \mu_o \text{exp}\left[\frac{\rho}{\rho^*}\sum_{i=0}^{5}\sum_{j=0}^{4} b_{ij}\left(\frac{T^*}{T}-1\right)^i\left(\frac{\rho}{\rho^*}-1\right)^j\right]$$

with

$$\mu_o = 10^{-6}\left(\frac{T}{T^*}\right)^{1/2}\left[\sum_{k=0}^{3} a_k\left(\frac{T^*}{T}\right)^k\right]^{-1}$$

where μ is the viscosity in Pa-s, ρ is the density in kg/m^3, T is the temperature in Kelvins, and the constants ρ and T are

$\rho^* = 317.763$ kg/m^3,

$T^* = 647.27$ K,

The coefficients in the expression for μ_o are:

$a_0 = 0.0181583$

$a_1 = 0.0177624$

$a_2 = 0.0105287$

$a_3 = -0.0036744$

and the values for b_{ij} are given in Table 7.7.3.

Table 7.7.3: Numerical Values of the Coefficients b_{ij}

	i = 0	i = 1	i = 2	i = 3	i = 4	i = 5
j = 0	0.5601938	0.162888	-0.130356	0.907919	-0.551119	0.146543
j = 1	0.235622	0.789393	0.673665	1.207552	0.0670665	-0.0843370
j = 2	-0.274637	-0.743539	-0.959456	-0.687343	-0.497089	0.195286
j = 3	0.145831	0.263129	0.346247	0.213486	0.100754	-0.032932
j = 4	-0.0270448	-0.0253093	-0.026776	-0.0822904	0.0602253	-0.0202595

Correlations for the enthalpy of saturated liquid water and saturated steam are taken from the RETRAN-02 code documentation *[7-14]*. The specific enthalpy of liquid water is given by

$$h_g = \begin{cases} \sum_{i=0}^{8} \text{CF1}_i \left[\ln(P)\right]^i & \text{for } 0.1 \text{ psia} \le P \le 950 \text{ psia} \\ \sum_{i=0}^{8} \text{CF2}_i \left[\ln(P)\right]^i & \text{for } 950 \text{ psia} \le P \le 2550 \text{ psia} \\ \sum_{i=0}^{8} \text{CF3}_i \left[(P_{CRIT}-P)^{0.41}\right]^i & \text{for } 2550 \text{ psia} < P \le P_{CRIT} \end{cases}$$

and the specific enthalpy of saturated steam is given by

$$h_g = \begin{cases} \sum_{i=0}^{11} \text{CG1}_i \left[\ln\left(P\right)\right]^i & \text{for } 0.1 \text{ psia} \leq P \leq 1500 \text{ psia} \\ \sum_{i=0}^{8} \text{CG2}_i \left[\ln\left(P\right)\right]^i & \text{for } 1500 \text{ psia} \leq P \leq 6550 \text{ psia} \\ \sum_{i=0}^{6} \text{CG3}_i \left[\left(P_{CRIT} - P\right)^{0.41}\right]^i & \text{for } 2650 \text{ psia} < P \leq P_{CRIT} \end{cases}$$

where h_f and h_g are the specific enthalpy in units of BTU/lbm, P is the pressure in psia, P_{CRIT} is the critical pressure (3208.2psia), and the constant coefficients are given in Table 7.7.4. Expressions for the temperatures of subcooled water superheated steam as functions of pressure and enthalpy are taken from RETRAN-02 *[7-14]*:

$$T_l = \sum_{i=0}^{i=i} \sum_{j=0}^{j=3} \text{CT1}_{i,j} P^{i,j}$$

and

$$T_v = \sum_{i=0}^{i=4} \sum_{j=0}^{j=4} \text{CT3}_{i,j} P^{i,j}$$

where T_l and T_v are the subcooled water and superheated steam temperatures in degrees Fahrenheit, P is the pressure in psia, h is the enthalpy in BTU/lbm, and the constant coefficients CT1 and CT3 are given in Table 7.7.5. The specific heats at constant pressure for subcooled water and superheated steam are calculated as the inverses of the partial derivations of the expressions for T_l and T_v with respect to enthalpy. The saturation temperature is obtained from the expression for T_l evaluated at the ambient pressure and the saturated liquid water specific enthalpy at that pressure.

Correlations for the specific enthalpies of subcooled liquid water and superheated steam as functions of pressure and enthalpy are taken from RETRAN-02 *[7-14]*:

$$v_l = \exp\left[\sum_{i=0}^{2} \sum_{j=0}^{4} \text{CN1}_{i,j} P^{i,j}\right]$$

and

$$v_v = \sum_{i=-1}^{2} \sum_{j=0}^{2} \text{CN2}_{i,j} P^{i,j}$$

where v_l and v_v are the subcooled water and superheated steam specific volumes in $ft^3 \,/\, lbm$, P is the pressure in psia, h is the enthalpy in BTU/lbm, and the constant coefficients CN1 and CN2 are listed in Table 7.7.6. The satruated liquid water density is computed from the value for v_l at the ambient pressure and the saturated steam specific enthalpy at that pressure. Similarly the saturated steam density is obtained from v_v at the ambient pressure and the satruated steam specific enthalpy at that pressure.

Table 7.7.4: Constant Coefficients in Expressions for Saturated Liquid Water and Saturated Steam Enthalpies as Functions of Pressure

i	$CF1_i$	$CF2_i$	$CF3_i$
0	$.6970887859 \times 10^{2}$	$.8408618802 \times 10^{6}$	$.9060030436 \times 10^{3}$
1	$.3337529994 \times 10^{2}$	$.3637413208 \times 10^{6}$	$-.1426813520 \times 10^{0}$
2	$.2318240735 \times 10^{1}$	$-.4634506669 \times 10^{6}$	$.1522233257 \times 10^{1}$
3	$.1840599513 \times 10^{0}$	$.1130306339 \times 10^{6}$	$-.6973992961 \times 10^{0}$
4	$-.5245502284 \times 10^{-2}$	$-.4350217298 \times 10^{3}$	$.1743091663 \times 10^{0}$
5	$.2878007027 \times 10^{-2}$	$-.3898988188 \times 10^{4}$	$-.2319717696 \times 10^{-1}$
6	$.1753652324 \times 10^{-2}$	$.6697399434 \times 10^{3}$	$.1694019149 \times 10^{-2}$
7	$-.4334859629 \times 10^{-3}$	$-.4730726377 \times 10^{2}$	$-.6454771710 \times 10^{-4}$
8	$.3325699282 \times 10^{-4}$	$.1265125057 \times 10^{1}$	$.1003003098 \times 10^{-5}$
i	$CG1_i$	$CG2_i$	$CG3_i$
0	$.1105836875 \times 10^{4}$	$-.2234264997 \times 10^{7}$	$.9059978254 \times 10^{3}$
1	$.1436943768 \times 10^{2}$	$.1231247634 \times 10^{7}$	$.5561957539 \times 10^{1}$
2	$.8018288621 \times 10^{0}$	$-.1978847871 \times 10^{7}$	$.3434189609 \times 10^{1}$
3	$.1617232913 \times 10^{-1}$	$.1859988044 \times 10^{2}$	$-.6406390628 \times 10^{0}$
4	$-.1501147505 \times 10^{-2}$	$-.2765701318 \times 10^{1}$	$.5918579484 \times 10^{-1}$
5	$.0000000000 \times 10^{0}$	$.1036033878 \times 10^{4}$	$-.2725378570 \times 10^{-2}$
6	$.0000000000 \times 10^{0}$	$-.2143423131 \times 10^{3}$	$.5006336938 \times 10^{-4}$
7	$.0000000000 \times 10^{0}$	$.1690507762 \times 10^{2}$	
8	$.0000000000 \times 10^{0}$	$-.4864322134 \times 10^{0}$	
9	$-.1237675562 \times 10^{-4}$		
10	$.3004773304 \times 10^{-5}$		
11	$-.2062390734 \times 10^{-6}$		

Table 7.7.5: Constant Coefficients in Expressions for Temperature as a Function of Pressure and Specific Enthalpy

			$CT1_{i,j}$		
	i = 0	i = 1	i = 2	i = 3	
j = 0	$0.3276275552 \times 10^{2}$	$0.9763617000 \times 10^{0}$	$0.1857226027 \times 10^{-3}$	$-0.4682674330 \times 10^{-6}$	
j = 1	$0.3360880214 \times 10^{-2}$	$-0.5595281760 \times 10^{-4}$	$0.1618595991 \times 10^{-6}$	$-0.1180204381 \times 10^{-9}$	
			$CT3_{i,j}$		
	i = 0	i = 1	i = 2	i = 3	i = 4
j = 0	$-0.1179100862 \times 10^{5}$	$0.2829274345 \times 10^{2}$	$-0.2678181564 \times 10^{-1}$	$0.1218742752 \times 10^{-4}$	$-0.2092033147 \times 10^{-8}$
j = 1	$0.1256160907 \times 10^{3}$	$-0.3333448495 \times 10^{0}$	$0.3326901268 \times 10^{-3}$	$-0.1477890326 \times 10^{-6}$	$0.2463258371 \times 10^{-10}$
j = 2	$-0.1083713369 \times 10^{0}$	$0.2928177730 \times 10^{-3}$	$-0.2972436458 \times 10^{-6}$	$0.1342639113 \times 10^{-9}$	$-0.2275585718 \times 10^{-13}$
j = 3	$0.3278071846 \times 10^{-4}$	$-0.8970959364 \times 10^{-7}$	$0.9246248312 \times 10^{-10}$	$-0.4249155515 \times 10^{-13}$	$0.7338316751 \times 10^{-17}$
j = 4	$-0.3425564927 \times 10^{-8}$	$0.9527692453 \times 10^{-11}$	$-0.1001409043 \times 10^{-13}$	$0.4703914404 \times 10^{-17}$	$-0.8315044742 \times 10^{-21}$

Table 7.7.6: Constant coefficients in Expressions for Specific Volume as a Function of Pressure and Specific Enthalpy

			$CN2_{i,j}$		
	i = 0	i = 1	i = 2	i = 3	i = 4
j = 0	$-0.4117961750\times 10^{1}$	$-0.3811294543\times 10^{-3}$	$0.4308265942\times 10^{-5}$	$-0.9160120130\times 10^{-8}$	$0.8017924673\times 10^{-11}$
j = 1	$-0.4816067020\times 10^{-5}$	$0.7744786733\times 10^{-7}$	$-0.6988467605\times 10^{-9}$	$0.1916720525\times 10^{-11}$	$-0.1760288590\times 10^{-14}$
j = 2	$-0.1820625039\times 10^{-8}$	$0.1440785930\times 10^{-10}$	$-0.2082170753\times 10^{-13}$	$-0.3603625114\times 10^{-16}$	$0.7407124321\times 10^{-19}$
			$CN2_{i,j}$		
	i = 0	i = 1	i = 2		
j = -1	$-0.1403086182\times 10^{4}$	$0.1802594763\times 10^{1}$	$-0.2097279215\times 10^{-3}$		
j = 0	$0.3817195017\times 10^{0}$	$-0.5394444747\times 10^{-3}$	$0.1855203702\times 10^{-6}$		
j = 1	$-0.6449501159\times 10^{-4}$	$0.8437637660\times 10^{-7}$	$-0.2713755001\times 10^{-10}$		
j = 2	$0.7823817858\times 10^{-8}$	$-0.1053834646\times 10^{-10}$	$0.3629590764\times 10^{-14}$		

Part III: Fuel Performance

CHAPTER

EIGHT

DEFORM-4: STEADY-STATE AND TRANSIENT PRE-FAILURE PIN BEHAVIOR

8.1 Overview and General Background

8.1.1 Objectives, General Background, and General Physical Description

The response of LMFBR fuel pins to transient accident conditions is an important safety concern. For transients leading to pin failure, the failure modes and initial fuel disruption depend in part on pre-transient irradiation effects, such as restructuring, fission-gas retention, fuel-cladding gap, central void size and makeup. As the transient fuel-pin models develop, an increasingly rigorous pre-transient fuel-pin characterization of the fuel pin is also required. For this reason, an effort has been made to integrate a detailed treatment of the pre-transient fuel-pin characterization into SAS4A. At the same time, an attempt has been made to assure that the models are consistent with the transient calculation and, where possible, to develop models in such a manner that they can be used in both the pre-transient and transient calculations.

Because the phenomena affecting fuel-pin integrity are not all well understood, most performance codes use one of two methods to predict the pin characterization: (1) empirical correlations derived from a data base of experimentally determined information, or (2) a phenomenological description of the process that contains parameters that need to be calibrated to experiments or the information data base. Care must be exercised with both approaches, but especially with the former method, because unrealistic values can be obtained if the pin conditions fall outside the correlation database. In the phenomenological (or mechanistic) approach the attempt is made to model the physical process taking place and then calibrate this to available data. Since the model attempts to describe the physical processes, it is possible to extrapolate the response to conditions outside the calibration data base with greater assurance than with the correlation approach. It was, therefore, decided that this mechanistic approach would be adopted wherever possible in the DEFORM-4 module of SAS4A.

This approach is different from that used by earlier versions of the SAS codes [8-1 - 8-4]. The former versions contained a very brief steady-state calculation of one time step at constant power and used correlations to determine the parameters necessary to start the transient calculation. The DEFORM-4 module simulates the pre-transient irradiation with a series of power changes and power levels of various time lengths. This provides the means to account for the reactor operating history and its effect on the physical state of the fuel pin, and leads to a more realistic description of the fuel pins subjected to the transient.

The pin is divided into a number of axial segments (≤24) of arbitrary length. If either the PLUTO2 or LEVITATE failed pin modeling is used, then there will be restrictions on the "arbitrariness" of these lengths (see Chapter 14 and Chapter 16). The fuel and cladding are assumed to occupy the same axial segment. Figure 8.1.1 shows an example of the axial and radial discretization for a fuel pin with an upper fission-gas plenum. The DEFORM-4 module is concerned with the axial region containing the driver fuel, axial blankets, and the fission-gas plenum. The core region, i.e., driver and axial blanket fuel segments, is shown in Figure 8.1.2. Illustrated are the three fuel-cladding gap conditions considered: (1) no contact between the fuel and cladding, (2) the fuel

elastically straining the cladding, and (3) the fuel plastically straining the cladding. Also illustrated is a central cavity that formed in the hotter regions of the driver fuel.

The fuel in an axial segment is divided into a series of radial cells (≤11). The radial cell boundaries may be determined on the basis of equal mass in each, except the inner and outer cells which contain half the mass of a regular cell, or with each cell thickness being equal, again except for the inner and outer cell which have half the nominal thickness (see Figure 8.1.3). The cladding is divided into two radial cells. The relationship between the general SAS4A cell structure and temperature locations and those used in DEFORM-4 is indicated in Figure 8.1.4. In the fuel region, the temperatures used by some DEFORM-4 calculations (refer to the specific models described below) are obtained by averaging the two SAS4A temperatures on either side of the radial cell boundary. Where the property being considered is an average over the annular cell, such as porosity, modulus of elasticity, conductivity, etc., the actual temperatures calculated by the thermal models in SAS4A are used.

A number of phenomena are treated in the pre-transient irradiation calculation. These are shown in Figure 8.1.5. Detailed descriptions are given in the following sections, but a brief outline is presented here. These major models include:

1. As-fabricated porosity migration by vapor transport (Section 8.3.1), which is responsible for the formation of the central void and provides a radial distribution of the remaining as-fabricated porosity that affects thermal conductivity;
2. Grain growth (Section 8.3.2), which affects the fission-gas release and fuel creep characteristics;
3. Fission-gas release (Section 8.3.3), which affects the radial distribution of total porosity and fission-gas-bubble-induced fuel swelling;
4. Fission-product swelling (Section 8.3.4), which includes solid fission product and fission-gas bubble swelling, and affects the radial porosity profile and fuel dimensions; and
5. Irradiation-induced cladding swelling (Section 8.3.5), which affects the cladding dimensions and density.

Since the transient calculation covers seconds and minutes rather than days and years, as in the pre-transient irradiation, the phenomena listed above are not considered, except for the fission-product swelling. Transient fission-gas release is assumed to occur only on fuel melting and is treated in the molten cavity routine, Section 8.3.7.

The thermoelastic mechanical calculations for the fuel and cladding are identical in both the pre-transient and transient. The cladding is treated as an elastic-perfectly-plastic material, although one of the options for the flow stress is dependent on strain and strain rate, and this introduces a type of work-hardening effect. In addition, the cladding is allowed to plasticly creep in response to temperature and stress conditions. Axial and radial deformations result from thermal expansion and mechanical interactions. The effects of fuel-cladding interaction are also considered in the fission product swelling calculation.

The fuel is allowed to crack radially whenever the circumferential stress exceeds the temperature-dependent fracture strength. The crack volume varies due to thermal and swelling effects. In the transient, the volume associated with the cracks, the fission gas, and the remaining as-fabricated porosity can be important in accommodating the thermal expansion on melting and determining the molten cavity pressure.

Fuel-pin failure can be initiated by a number of criteria. These include the following;

1. Time,
2. Fuel temperature,
3. Melt fraction,
4. Molten cavity pressure,
5. Cladding stress,
6. Eutectic penetration for U-5FS fuel,

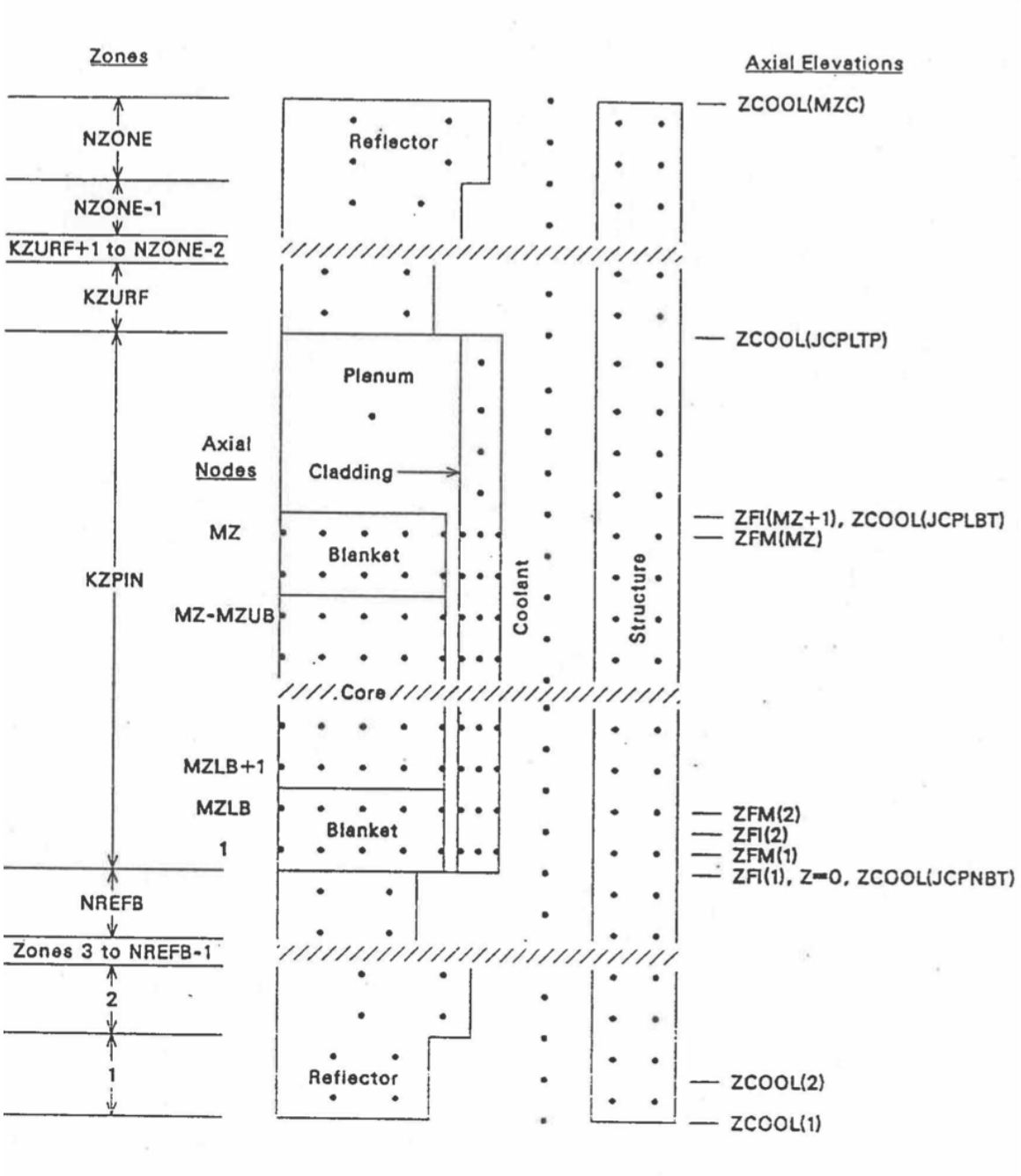


Figure 8.1.1: Schematic of SAS4A Channel Discretization

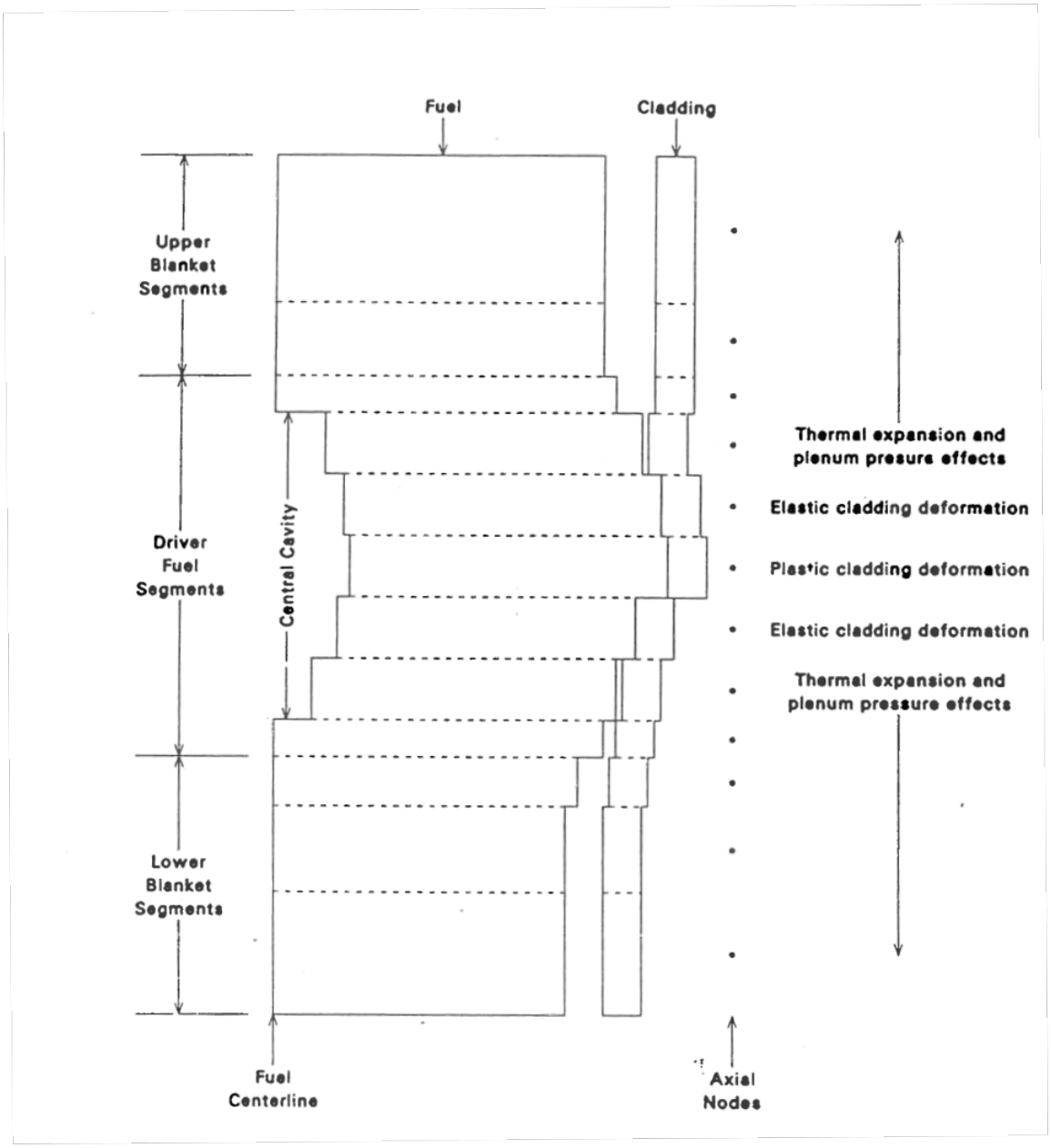


Figure 8.1.2: DEFORM-4 Axial Segmentation with Possible Fuel-cladding Interactions

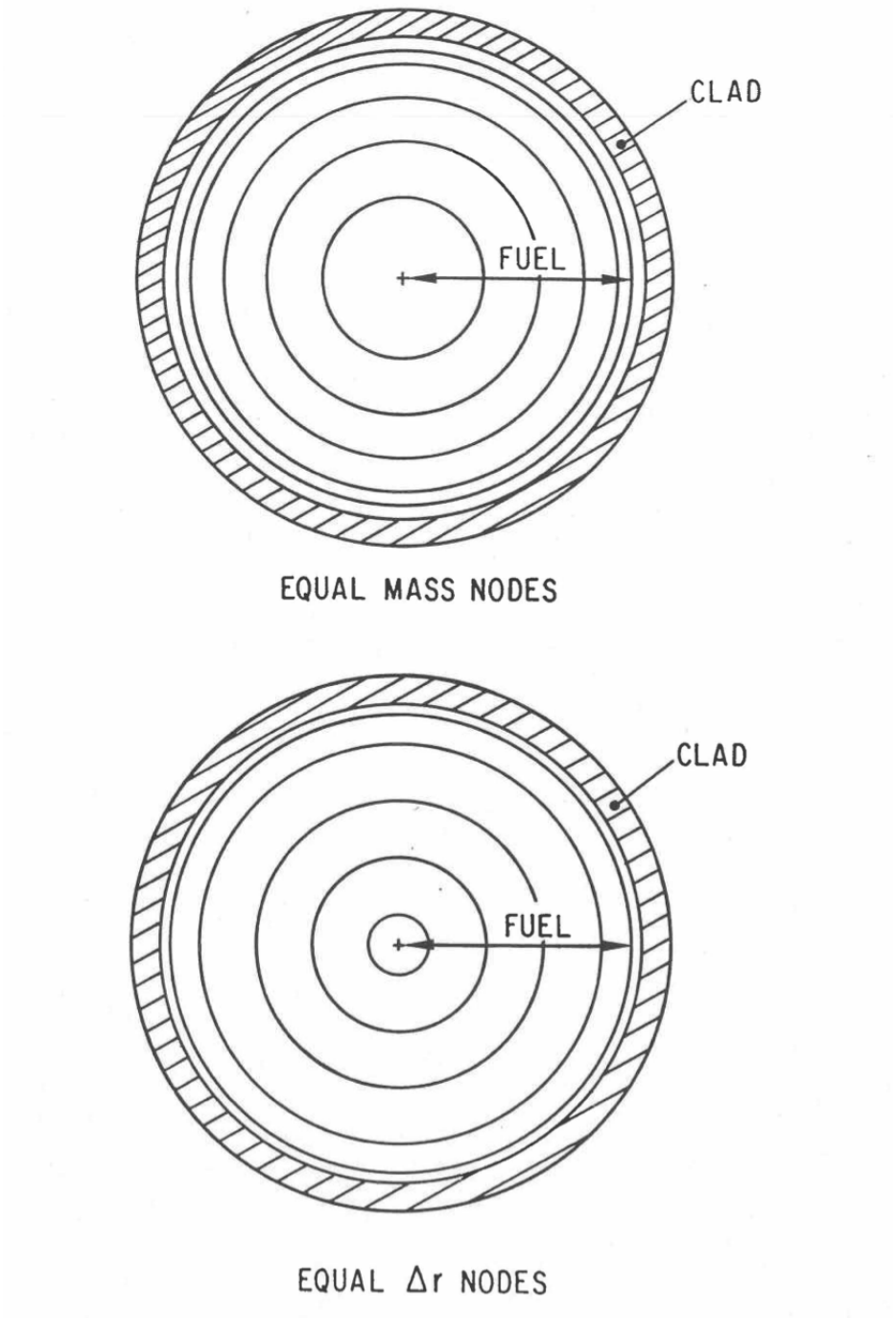


Figure 8.1.3: Radial Cell Construction Options

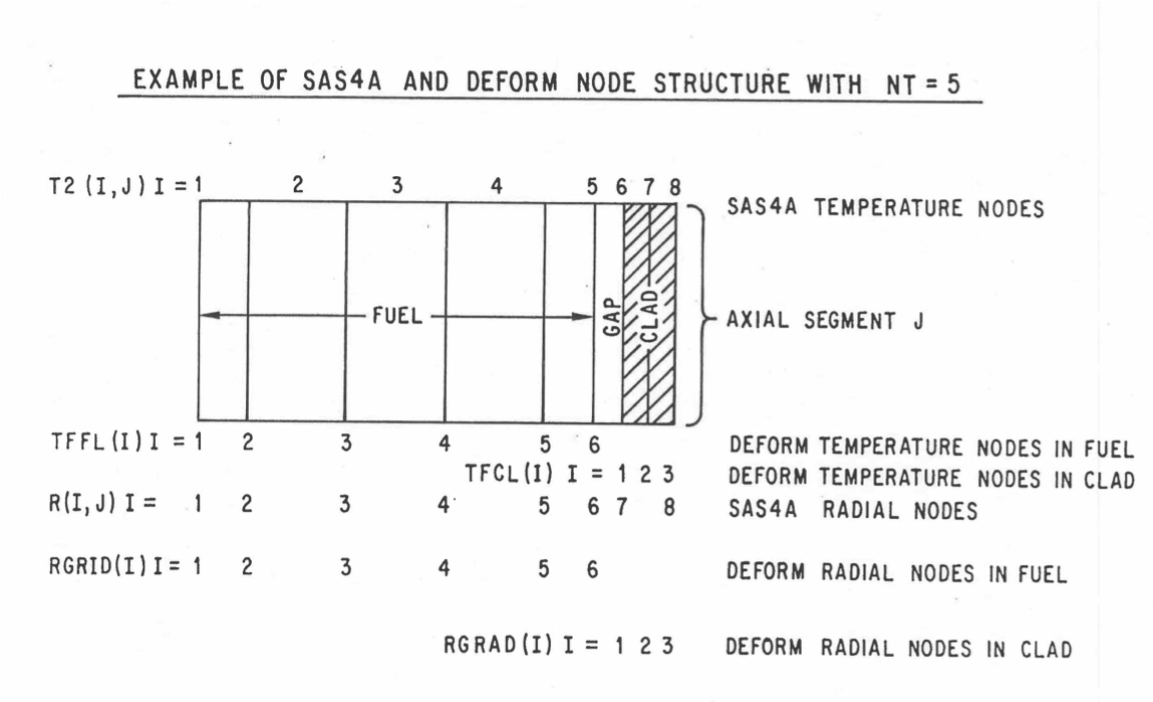


Figure 8.1.4: Relationship Between SAS4A and DEFORM-4 Cells and Cell-boundaries

7. Cladding reaching conditions equivalent to PLUTO2/LEVITATE failure propagation model,
8. Eutectic penetration with cladding stress for metal fuel gauge,
9. Melt fraction for time and failure propagation for location.

Also included are a number of life-fraction correlations based on Larson-Miller and Dorn parameters, but these have not been tied into the automatic failure initiation. The use of these to initiate failure is through observation of the life fractions and then selection of a time and location to be used in a subsequent restart calculation.

8.1.2 Interaction with other SAS4A Models

Because the DEFORM-4 module performs all the fuel-pin characterization and mechanics calculations for SAS4A, it must be capable of exchanging data with the rest of the code. This is done through the use of COMMON blocks accessible by other modules. In the pre-transient and pre-failure transient calculations, DEFORM-4 communicates with the thermal hydraulic models in SAS4A and PINACLE. Once failure has occurred and CLAP, PLUTO2, or LEVITATE become activated, DEFORM-4 is no longer used for that channel, but still performs the calculations for any remaining unfailed channels.

In the initiation of CLAP, LEVITATE, or PLUTO2, the main responsibility of DEFORM-4 is to assure that the necessary information is available for these modules to continue the calculation. If the user has so specified by appropriate input options, then DEFORM-4 will determine when the failure has occurred and initiate the activation of the appropriate module for fuel motion. If it is not being used, failure can still be initiated, but important parameters, such as fission gas distributions and molten cavity pressures, will not be available for PLUTO2 and LEVITATE. This will cause significant problems in the computations carried out by these modules. The CLAP module does not depend on DEFORM-4 results in its calculations for cladding motion, except for the initial pin dimensions.

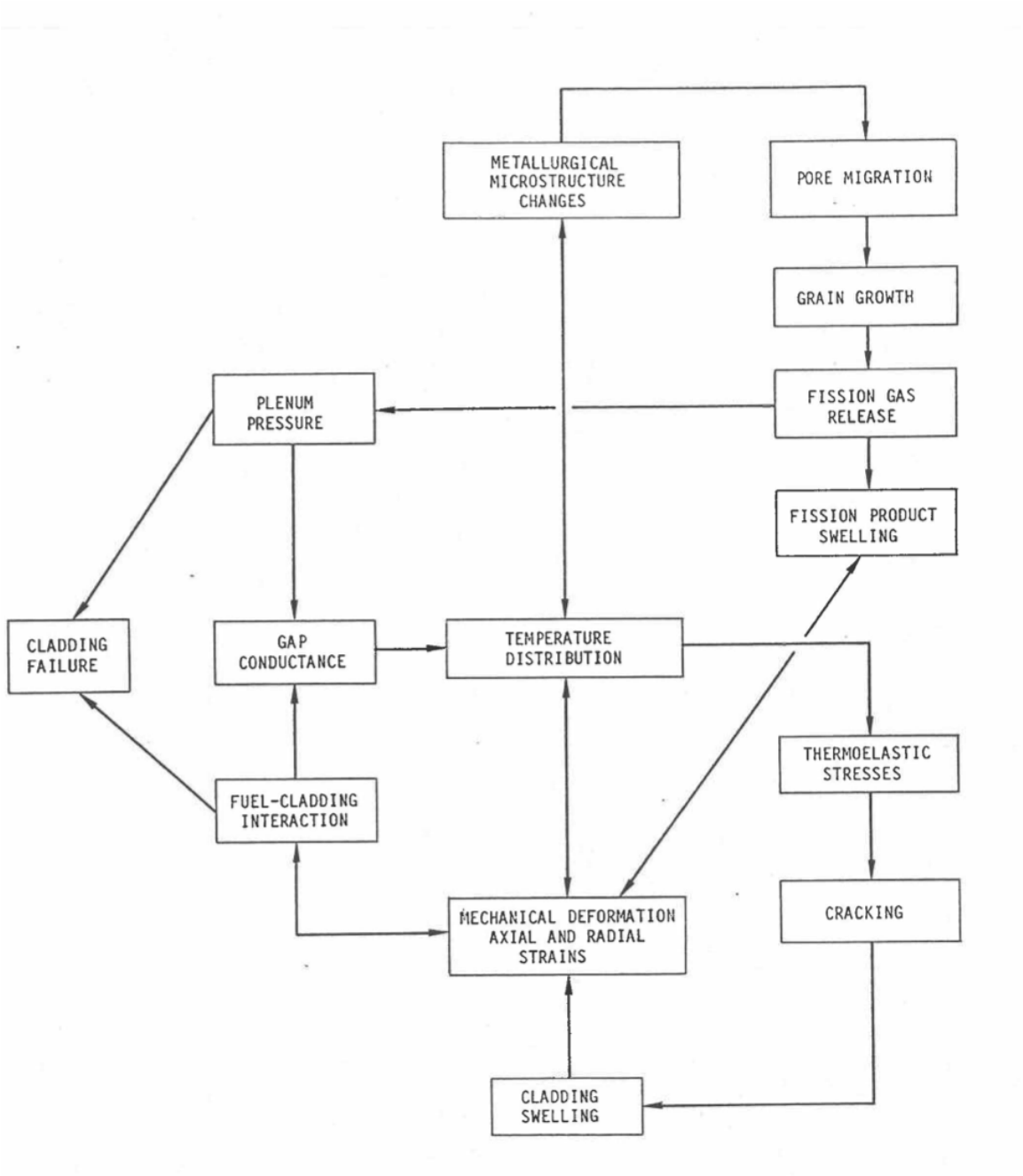


Figure 8.1.5: DEFORM-4 Mechanical and Phenomenological Considerations and Their Interactions

The main communication with the rest of SAS4A on an interactive basis is through the core temperatures and pressures supplied by the thermal hydraulic models and then the radii, axial lengths, porosity distributions which affect thermal conductivities, and gap conductances which are returned from DEFORM-4. In order to avoid time-consuming iterations between the thermal hydraulic models and DEFORM-4, the calculations are performed serially. The thermal hydraulic models use the geometry they have at the beginning of a time step with the power, flow, and time-step length to determine temperatures at the end of the time step (see Chapter 3). These final temperatures and the initial temperatures are then used by DEFORM-4 to determine the thermal mechanical changes that take place during the same interval. The new conditions are then transferred back to the thermal hydraulic routines as the initial conditions for the next time step.

This method of interaction with the SAS4A thermal hydraulic models has been employed to avoid the resource-consuming iterations that could occur between the thermal hydraulic and pin characterization routines. These iterations would assure complete consistency between the temperatures and the characterization state of the pin, but at a price in computational effort that would preclude the use of the code for many of the multiple channel, extended pre-transient and transient cases of interest in initiating-phase accident analysis. Instead of iterations, there is control over the length of the computational time step. In the early pre-transient irradiation where pore migration and initial fission-gas release cause changes in the geometry and heat-transfer properties that would greatly affect the temperatures, time intervals on the order of one to two days should be used. During power changes it may be necessary to cut these further, especially during the initial startup to full power. In later stages, the time steps can be on the order of 10 to 30 days. During the transient where the time steps are controlled by a variety of restrictions, such as maximum reactivity change, maximum temperature change, etc., the heat-transfer time steps are small enough, on the order of less than a second, to remove any problems with inconsistencies. DEFORM-4 would be able to handle considerably longer time steps without problems. In practice, these types of limits have resulted in excellent results with minimum computational effort.

8.2 Fuel-pin Mechanics

In the treatment adopted in DEFORM-4, the fuel pin in an axial segment is divided into 6 radial zones, not all of which need exist. These are (1) the central void, (2) the molten fuel zone, (3) the solid, continuous fuel zone, (4) the cracked fuel zone, (5) the fuel-cladding gap, and (6) the fuel-pin cladding. The zones are illustrated in Figure 8.2.1. Each zone may consist of one or more cells. These zones will be explained in detail in the following sections.

The approach used is to divide the calculation into the thermoelastic solution, and then superimpose on this the plastic deformation resulting from fuel swelling or cladding stress induced plastic creep and irradiation swelling

$$\varepsilon_T = \varepsilon_e + \varepsilon_{th} + \varepsilon_s \tag{8.2-1}$$

where

ϵ_r = Total strain at the cell boundary

ϵ_e = Elastic strain from applied boundary forces

ϵ_{th} = Thermal expansion induced strain

ϵ_s = Swelling strains from the solid and volatile fission products in the fuel, or irradiation induced void formation and stress induced plastic creep in the cladding.

These terms are discussed in the following sections.

The cladding is assumed to be an elastic-perfectly-plastic material. Several functions are available as options to provide the flow stress of the cladding. Once the fuel-cladding interface pressure produces a circumferential stress exceeding the flow stress, the interface pressure is limited to that necessary to achieve the flow stress and the cladding will follow the fuel deformation until the conditions bring the cladding back into the elastic behavior

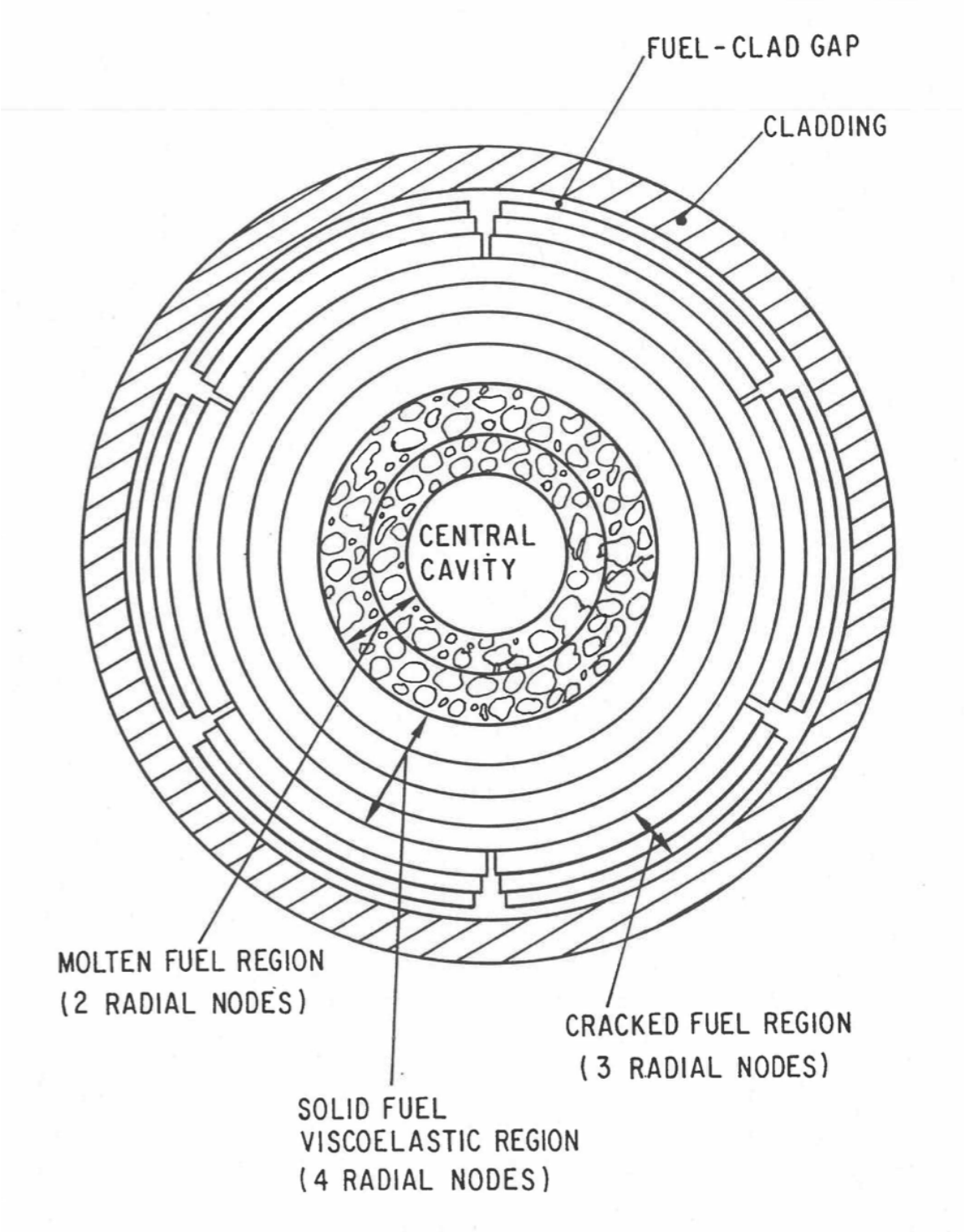


Figure 8.2.1: DEFORM-4 Radial Zones

region. One of the options for flow stress includes the effects of previous strain, strain rate, temperature, and irradiation history. When this option is used, the flow stress changes as these parameters change, providing a work hardening and strain-rate dependence. Besides this perfectly plastic behavior, there is plastic creep of the cladding at conditions below the flow stress. This strain is calculated and added to the accumulated strain.

8.2.1 Solid Fuel and Cladding Thermoelastic Solution

In the solid fuel and cladding, the material is assumed to be continuous, isotropic, elastic, and axisymmetric. Because of the axisymmetry, all shear stresses and strains are assumed to be zero. The generalization of Hooke's Law to three dimensions is used to provide the linear elastic relationship between the stresses and strains. The thermal expansion strains are included through the principle of superposition of linear equations. These considerations therefore yield the following set of constitutive equations.

$$\varepsilon_{\mathrm{r}} = \frac{1}{E}\left[\sigma_{\mathrm{r}} - \nu\left(\sigma_{\theta} + \sigma_{\mathrm{z}}\right)\right] + \Delta\left(\alpha T\right) \tag{8.2-2}$$

$$\varepsilon_{\theta} = \frac{1}{E}\left[\sigma_{\theta} - \nu\left(\sigma_{\mathrm{r}} + \sigma_{\mathrm{z}}\right)\right] + \Delta\left(\alpha T\right) \tag{8.2-3}$$

$$\varepsilon_{\mathrm{z}} = \frac{1}{E}\left[\sigma_{\mathrm{z}} - \nu\left(\sigma_{\mathrm{r}} + \sigma_{\theta}\right)\right] + \Delta\left(\alpha T\right) \tag{8.2-4}$$

$$\Delta\left(\alpha T\right) = \alpha\left(T_2\right)\left(T_2 - T_{\mathrm{r}}\right) - \alpha\left(T_1\right)\left(T_1 - T_{\mathrm{r}}\right) \tag{8.2-5}$$

where

ε_{r}, ε_{θ}, ε_{z} = Strain in the radial, circumferential, and axial directions, respectively

σ_{r}, σ_{θ}, σ_{z} = Stress in the radial, circumferential, and axial directions, respectively

υ = Poisson's ratio for the material

$\alpha(T)$ = Mean linear thermal expansion coefficient

E = Modulus of elasticity

T_2, T_1 = Temperature at the final and initial states, respectively,

T_{r} = Reference temperature

The strains are related to the displacements through geometrical considerations. In the cylindrical coordinate system used in SAS4A, these kinematic equations are as follows:

$$\varepsilon_{\mathrm{r}} = \frac{\mathrm{du}}{\mathrm{dr}} \tag{8.2-6}$$

$$\varepsilon_{\theta} = \frac{u}{r} + \frac{1}{r}\frac{\mathrm{dv}}{\mathrm{d}\theta} \tag{8.2-7}$$

$$\varepsilon_{\mathrm{z}} = \frac{\mathrm{dw}}{\mathrm{dz}} \tag{8.2-8}$$

where

u = Displacement in the radial direction r

υ = Displacement in the circumferential direction θ

w = Displacement in the axial direction z

Since material is assumed to be axisymmetric, there is no variation of v circumferentially, so Eq. (8.2-7) reduces to

$$\varepsilon_{\theta} = \frac{u}{r} \tag{8.2-9}$$

In order to be able to obtain simple analytical solutions with the above equations, a generalized plane strain approximation has been employed. Each axial segment is assumed to elongate uniformly over the cross section to maintain a plane interface between segments. Eq. (8.2-8) can therefore be rewritten as

$$\varepsilon_z = z_\mathrm{o} \tag{8.2-10}$$

where

z_o = Axial plane strain for the segment

Since the cells under consideration are assumed to be at rest, with no shear stresses, mechanical considerations provide the following equation of equilibrium.

$$\frac{\mathrm{d}\sigma_\mathrm{r}}{\mathrm{dr}} + \frac{\sigma_\mathrm{r} - \sigma_\theta}{r} = 0 \tag{8.2-11}$$

Eq. (8.2-2) through Eq. (8.2-6), and Eq. (8.2-9) through Eq. (8.2-11) form the set of equations solved by DEFORM-4 for the thermoelastic response. Eq. (8.2-10) is substituted into Eq. (8.2-4) and this is solved for the axial stress, σ_z. This result is then substituted along with Eq. (8.2-6) and Eq. (8.2-9) in Eq. (8.2-2) and Eq. (8.2-3) to yield

$$\sigma_\mathrm{r} = \frac{E}{(1+\nu)(1-2\nu)}\left[(1-\nu)\frac{\mathrm{du}}{\mathrm{dr}} + \nu\frac{u}{r} + \nu z_\mathrm{o} - (1+\nu)\Delta(\alpha T)\right] \tag{8.2-12}$$

$$\sigma_\theta = \frac{E}{(1+\nu)(1-2\nu)}\left[\nu\frac{\mathrm{du}}{\mathrm{dr}} + (1-\nu)\frac{u}{r} + \nu z_\mathrm{o} - (1+\nu)\Delta(\alpha T)\right] \tag{8.2-13}$$

The stresses are expressed in terms of the radial displacement function $u(r)$. When these equations are used in the equilibrium Eq. (8.2-11), the following is obtained, assuming the modulus of elasticity, E, is constant over the region of interest. The value for the modulus of elasticity is the mass-weighted average of all those cells in this zone.

$$\frac{1}{r}\frac{\mathrm{d}}{\mathrm{dr}}\left(r\frac{\mathrm{du}}{\mathrm{dr}}\right) - \frac{u}{r^2} = \frac{(1+\nu)}{(1-\nu)}\frac{\mathrm{d}}{\mathrm{dr}}\left[\Delta(\alpha T)\right] \tag{8.2-14}$$

The solution to this differential equation may be obtained as

$$u(r) = \left(\frac{1+\nu}{1-\nu}\right) rI(r) + C_1 r + \frac{C_2}{r} \tag{8.2-15}$$

where

$C_1,\ C_2$ = Constants of integration

The function $I(r)$ is defined as

$$I(r) = \frac{1}{r^2}\int_\rho^r \Delta(\alpha T)\, r'\mathrm{dr}' \tag{8.2-16}$$

where

ρ = Inner radius of the zone under consideration

If Eq. (8.2-15) is used to rewrite Eq. (8.2-12) and Eq. (8.2-13), the radial and circumferential stresses as functions of r, C_1, and C_2 may be obtained.

$$\frac{\sigma_\mathrm{r}(r)}{E} = -\frac{I(r)}{(1-\nu)} + \frac{1}{(1+\nu)(1-2\nu)}\left[C_1 - \frac{(1-2\nu)C_2}{r^2} + \nu z_0\right] \tag{8.2-17}$$

$$\frac{\sigma_\theta(r)}{E} = \frac{I(r)}{(1-\nu)} + \frac{1}{(1+\nu)(1-2\nu)}\left[C_1 - \frac{(1-2\nu)C_2}{r^2} + \nu z_0\right] - \left(\frac{\Delta\alpha T}{(1-\nu)}\right) \tag{8.2-18}$$

The constants of integration, C_1 and C_2, may be determined by the following boundary conditions:

$$\sigma_{\mathrm{r}}(r=\rho)=\sigma_{\rho} \tag{8.2-19}$$

$$\sigma_{\mathrm{r}}(r=\eta)=\sigma_{\eta} \tag{8.2-20}$$

where

η = Outer radius of the zone under consideration

σ_{ρ} = Stress at the inner surface of the zone

σ_{η} = Stress at the outer surface of the zone

For the fuel, ρ and η would correspond to the inner solid and outer uncracked cell boundaries, respectively. For the cladding, they would correspond to the inner and outer surfaces, respectively.

The constants may then be determined by substituting Eq. (8.2-19) and Eq. (8.2-20) into Eq. (8.2-17) and Eq. (8.2-18) and solving them simultaneously for C_1 and C_2.

$$C_1=\frac{(1+v)(1-2v)}{(\eta^2-\rho^2)}\left[\frac{I(\eta)\eta^2}{(1-v)}-\frac{1}{E}\rho\left(\rho^2\sigma_{\rho}-\eta^2\sigma_{\eta}\right)\right]-vz_{\mathrm{o}} \tag{8.2-21}$$

$$C_2=\frac{(1+\nu)\rho^2\eta^2}{(\eta^2-\rho^2)}\left[\frac{I(\eta)}{(1-\nu)}-\frac{(\sigma_{\rho}-\sigma_{\eta})}{E}\right] \tag{8.2-22}$$

These constants may then be used in Eq. (8.2-15), Eq. (8.2-17), and Eq. (8.2-18) to determine the radial displacement and stresses due to externally applied forces and thermal expansion.

$$u(r)=u_{\mathrm{f}}(r)+u_{\mathrm{th}}(r) \tag{8.2-23}$$

$$\begin{aligned} u_{\mathrm{f}}(r) &= \text{ Elastic displacement due to externally applied stress } \sigma_{\rho} \text{ and } \sigma_{\eta} \\ &= \frac{(1+\nu)}{(\eta^2-\rho^2)E}\left\{\eta^2\sigma_{\eta}\left[(1-2\nu)r+\frac{\rho^2}{r}\right]-\rho^2\sigma_{\rho}\left[(1-2\nu)r+\frac{\eta^2}{r^2}\right]\right\}-\nu z_{\mathrm{o}}^{f}r \end{aligned} \tag{8.2-24}$$

$$\begin{aligned} u_{\mathrm{th}}(r) &= \text{ Displacements due to thermal expansion} \\ &= \left(\frac{1+\nu}{1-\nu}\right)\left\{rI(r)+\frac{I(\eta)\eta^2}{(\eta^2-\rho^2)}\left[(1-2\nu)r+\frac{\rho^2}{r}\right]\right\}-\nu z_{\mathrm{o}}^{th}r \end{aligned} \tag{8.2-25}$$

where z_{o} has been divided into its thermal and boundary force components, z_{o}^{th} and z_{o}^{f} respectively (see Section 8.2.4).

The stresses are similarly divided into their thermal and externally applied force components.

$$\sigma_{\mathrm{r}}(r)=\sigma_{\mathrm{r}}^{f}(r)+\sigma_{\mathrm{r}}^{\mathrm{th}}(r) \tag{8.2-26}$$

$$\begin{aligned} \sigma_{\mathrm{r}}^{f}(r) &= \text{ Radial stress from externally applied forces} \\ &= \frac{1}{(\eta^2-\rho^2)}\left[\eta^2\sigma_{\eta}\left(1-\frac{\rho^2}{r^2}\right)-\rho^2\sigma_{\rho}\left(1-\frac{\eta^2}{r}\right)\right] \end{aligned} \tag{8.2-27}$$

$$\begin{aligned} \sigma_{\mathrm{r}}^{\mathrm{th}}(r) &= \text{ Radial stress from thermal expansion} \\ &= \frac{E}{(1-\nu)}\left[\frac{\eta^2 I(\eta)}{(\eta^2-\rho^2)}\left(1-\frac{\rho^2}{r^2}\right)-I(r)\right] \end{aligned} \tag{8.2-28}$$

$$\sigma_{\theta}(r)=\sigma_{\mathrm{r}}^{f}(r)=\sigma_{\mathrm{r}}^{th}(r) \tag{8.2-29}$$

$$\begin{aligned} \sigma_{\theta}^{f}(r) &= \text{ Circumferential stress from externally applied forces} \\ &= \frac{1}{(\eta^2-\rho^2)}\left[\eta^2\sigma_{\eta}\left(1+\frac{\rho^2}{r^2}\right)-\rho^2\sigma_{\rho}\left(1+\frac{\eta^2}{r^2}\right)\right] \end{aligned} \tag{8.2-30}$$

$$\begin{aligned}\sigma_{\mathrm{r}}^{th}(r) = & \text{ Circumferential stress from thermal expansion} \\ & = \frac{E}{(1-\nu)}\left[\frac{\eta^2 I(\eta)}{(\eta^2-\rho^2)}\left(1+\frac{\rho^2}{r^2}\right)+I(r)-\Delta(\alpha T)\right]\end{aligned} \tag{8.2-31}$$

Throughout the derivation given above, only the thermal components contain the explicit reference to a change from one state to another, i.e., reliance on temperatures T_2 and T_1 of Eq. (8.2-5). All the force components are based on a change from a zero stress state to some state created by the imposition of the external force boundary conditions. In the calculational procedure used in SAS4A, there are a series of time steps with different conditions at the beginning and end of the step. As mentioned above, the SAS4A thermal hydraulic routines generate temperatures at the beginning and end of the time step and then DEFORM-4 determines the changes in dimensions, stress state, and characterization that occur during the time step. In order to use this natural, incremental approach, Eq. (8.2-24), Eq. (8.2-27), and Eq. (8.2-30) must be changed to represent the changes from the beginning of the time step to the end.

Eq. (8.2-27) gives the radial stress state set up by the externally applied stresses σ_η and σ_ρ when the material was initially in an unstressed condition. If a different set of conditions existed, then a different stress state is set up. The change from one state to the next can be found by using Eq. (8.2-27) twice, once with each external stress boundary condition, and then subtracting.

$$\Delta\sigma_{\mathrm{r,i}}^{f}(r) = \frac{1}{(\eta^2-\rho^2)}\left[\eta^2\left(1-\frac{\rho^2}{r^2}\right)(\sigma_{\eta,\mathrm{i}}-\sigma_{\eta,\mathrm{i}-1})-\rho^2\left(1-\frac{\eta^2}{r^2}\right)(\sigma_{\rho,\mathrm{i}}-\sigma_{\rho,\mathrm{i}-1})\right] \tag{8.2-32}$$

where

$\sigma_{\rho,\mathrm{i}}$, $\sigma_{\eta,1}$ = Inner and outer externally applied stresses at the end of time step i, respectively

$\sigma_{\rho,\mathrm{i}-1}\sigma_{\eta,\mathrm{i}-1}$ = Inner and outer externally applied stresses at the beginning of time step i (end of time step $i-1$), respectively

$\Delta\sigma_{\mathrm{r,i}}^{f}(r)$ = Incremental change in the radial stress due to changes in the externally applied stresses σ_ρ and σ_η

In a similar manner, Eq. (8.2-24) and Eq. (8.2-30) can be used to provide the changes occurring during the computational time step.

$$\begin{aligned}\Delta u_{\mathrm{f,i}}(r) = \frac{(1+v)}{(\eta^2-\rho^2)E}\Bigg\{&\eta^2(\sigma_{\mathrm{n,j}}-\sigma_{\eta,i-1})\left[(1-2\nu)r+\frac{\rho^2}{r}\right] \\ &-\rho^2(\sigma_{\rho,\mathrm{i}}-\sigma_{\rho\mathrm{i}-1})\left[(1-2\nu)r+\frac{\eta^2}{r}\right]\Bigg\}-\nu\Delta z_{\mathrm{o,i}}^{f}r\end{aligned} \tag{8.2-33}$$

$$\begin{aligned}\Delta\sigma_{\theta,\mathrm{i}}^{f}(r) = \frac{(1)}{(\eta^2-\rho^2)}\Bigg[&\eta^2\left(1+\frac{\rho^2}{r^2}\right)(\sigma_{\eta,\mathrm{i}}-\sigma_{\eta,\mathrm{i}-1}) \\ &-\rho^2\left(1+\frac{\eta^2}{r^2}\right)(\sigma_{\rho,\mathrm{i}}-\sigma_{\rho,\mathrm{i}-1})\Bigg]\end{aligned} \tag{8.2-34}$$

where

$\Delta u_{\mathrm{f,i}}(r)$ = Incremental elastic displacement due to changes in the externally applied stresses σ_ρ and σ_η

$\Delta\sigma_{\theta,\mathrm{i}}^{f}(r)$ = Incremental change in the circumferential stress due to changes in the externally applied stresses σ_ρ and σ_{v}

$\Delta z_{\mathrm{o,i}}^{f}$ = Incremental change in the axial plane strain due to changes in the external applied stresses σ_ρ and σ_η

As mentioned above, this same procedure is not required for the thermally induced stresses and strains, since they are derived from explicit temperature changes. With the initial and final temperatures in Eq. (8.2-5) defined as the temperatures at the beginning and end of the current time step, respectively, the incremental changes in the stresses and strains are determined.

Eq. (8.2-25), Eq. (8.2-28), and Eq. (8.2-31), define the incremental changes in the deformation, radial stress, and circumferential stress in response to a temperature change during the time step. Eq. (8.2-32) through Eq. (8.2-34) define the changes in response to changes in the externally applied forces. The incremental changes in the axial stresses can be found from Eq. (8.2-4), once the axial plane strain is determined. The separation of the stresses into the thermal and force components makes it possible to implement thermal stress relaxation in a straightforward manner.

The subroutine FSIGMA solves the equations for the fuel and the subroutine CSIGMA solves them for the cladding.

8.2.2 Cracked Fuel Thermoelastic Solution and Crack Volume

If the circumferential stress at a cell boundary exceeds the fracture strength of the fuel, the cell immediately inside that boundary is assumed to crack. The new outer solid fuel boundary is then studied to determine if it will also crack. This process is repeated until a stable solid boundary is reached, or cracking occurs to the central void or molten fuel boundary. This procedure is carried out in the subroutine FSIGMA. The treatment for the solid zone was presented previously and the solution in the cracked zone is given below.

In the cracked zone, it is assumed that numerous radial and transverse cracks exist and extend inward to the same radial position. Under these conditions the circumferential and axial stress, σ_θ and σ_z, respectively, are both set equal to the negative of the plenum gas pressure, P_g, since it is assumed that communication exists with the plenum.

$$\sigma_\theta = \sigma_z = -P_g \tag{8.2-35}$$

If these values are substituted into the equilibrium Eq. (8.2-11), and the integration performed from the outer fuel radius, R_f, to a radius r, the radial stress at any point in the cracked region results:

$$\sigma_r(r) = -P_g + \frac{R_f}{r}(P_g + \sigma_{FC}) \tag{8.2-36}$$

where

σ_{FC} = Fuel-cladding interface stress, the negative of the interface pressure

If the integration is performed outward from the boundary between the continuous and the cracked fuel, η, to a radius r, then the stress in the cracked fuel is of the form:

$$\sigma_r(r) = -P_g + \frac{\eta}{r}(P_g + \sigma_\eta) \tag{8.2-37}$$

where

σ_η = Stress at the outer boundary of the solid fuel

Since Eq. (8.2-36) and Eq. (8.2-37) must define the same stress at a given radius, the relationship between the stress at the solid-cracked boundary and the fuel-cladding interface may be determined as

$$\sigma_\eta = -P_g + \frac{R_f}{\eta}(P_g + \sigma_{FC}) \tag{8.2-38}$$

It should be noted that the radial stress in the cracked region contains no dependence on the thermal expansion of the region.

To obtain the radial displacement, u_r, Eq. (8.2-35) and Eq. (8.2-36) are substituted into Eq. (8.2-2) and the result used in Eq. (8.2-6). Upon integration, the radial displacement is given by

$$u_r = u_\eta + (2\nu - 1)(r - \eta)\frac{P_g}{E_c} + \frac{(P_g - \sigma_\eta)}{E_c}\eta \ln\left(\frac{r}{\eta}\right) + \int_\eta^r \Delta(\alpha T)\, dr' \tag{8.2-39}$$

where

u_{r} = Displacement at radius r in the cracked fuel zone

u_{η} = Displacement of the outer surface of the continuous fuel zone

E_{c} = Mass averaged modulus of elasticity in the cracked fuel zone

As in the section on solid fuel or cladding, the above derivations contain the implicit assumption that the cells start out in an unexpanded, stress free state. This is not what exists at the beginning of the time step. Eq. (8.2-35) through Eq. (8.2-39) can be used to obtain the changes that occur from the beginning of the time step to the end.

$$\Delta\sigma_{\theta,\mathrm{i}} = \Delta\sigma_{\mathrm{z,i}} = -\Delta P_{\mathrm{g,i}} \tag{8.2-40}$$

$$\begin{aligned}\Delta\sigma_{\mathrm{r,i}} &= -\Delta P_{\mathrm{g,i}} + \frac{R_{\mathrm{f}}}{r}\left(\Delta P_{\mathrm{g,i}} + \Delta\sigma_{\mathrm{FC,i}}\right) \\ &= -\Delta P_{\mathrm{g,i}} + \frac{\eta}{r}\left(\Delta P_{\mathrm{g,i}} + \Delta\sigma_{\mathrm{n,i}}\right)\end{aligned} \tag{8.2-41}$$

$$\Delta\sigma_{\mathrm{n,i}} = -\Delta P_{\mathrm{g,i}} + \frac{R_{\mathrm{f}}}{r}\left(\Delta P_{\mathrm{g,i}} + \Delta\sigma_{\mathrm{FC,i}}\right) \tag{8.2-42}$$

where

$\Delta P_{\mathrm{g,i}}$ = Change in plenum pressure during time step i

$\Delta\sigma_{\mathrm{FC,i}}$ = Change in fuel-cladding interface stress during time step i

$\Delta\sigma_{\mathrm{n,i}}$ = Change in stress at solid fuel-cracked fuel boundary during time step i

The displacement calculated in Eq. (8.2-39) contains two parts, that due to forces and that due to temperature changes. As with the solid fuel, the thermal effects are already handled in an incremental fashion, so no changes are required. However, the force effects have to be modified to handle the changes during the time step. Using the solid fuel results for incremental changes, the value of outer solid fuel displacement, u_{η}, will be an incremental change. The force effects can be made incremental in the manner used above.

$$u_{\mathrm{r,i}} = u_{\eta,\mathrm{i}}\left(2\nu - 1\right)\left(r - \eta\right)\frac{\Delta P_{\mathrm{g,i}}}{E_{\mathrm{c}}} + \frac{\left(\Delta P_{\mathrm{g,i}} + \Delta\sigma_{\mathrm{n,i}}\right)}{E_{\mathrm{c}}}\eta\ln\left(\frac{r}{\eta}\right) + \int_{\eta}^{r}\Delta\left(\alpha T\right)\mathrm{d}r' \tag{8.2-43}$$

Eq. (8.2-43) gives the displacements of the cracked fuel nodes in the incremental manner desired. These equations are solved in the subroutine FSIGMA.

As the dimensions of a cracked fuel cell change, so will the fraction of the volume that is associated with the cracks. These changes are calculated in the subroutine CRAKER. In the current version of DEFORM-4, the volume associated with transverse, or axial, cracking is neglected. The radial crack volume fraction is affected by three factors: (1) changes in the cell boundary locations, (2) circumferential strain, and (3) fission-product-induced fuel swelling. The first two processes are treated in the subroutine CRAKER, whereas the third is treated in the subroutine FSWELL.

$$\Delta V_{\mathrm{CRK}} = \Delta V\left(u\right)\epsilon_{\theta}^{c}\Delta V_{\mathrm{s}} \tag{8.2-44}$$

where

ΔV_{CRK} = Fractional volume change due to radial cracks

$\Delta V\left(u\right)$ = Fractional volume change associated with the changes in the radial boundaries of the cell by the constant displacement u

ϵ_{θ}^{c} = Circumferential strain of the cracked fuel cell

ΔV_{s} = Fractional volume change associated with fission-product swelling in the cell

The first term in Eq. (8.2-44) is based on purely geometric considerations. If an annulus with inner radius r_{i} and outer radius r_{o} is moved radially by an amount u, the new inner and outer radius become $r_{\mathrm{i}}+u$ and $r_{\mathrm{o}}+u$, respectively. This new annulus has a different volume from the original configuration. The change in volume fraction is assumed to be contained in the cracks in the fuel cell.

$$\Delta V\left(u\right)=\frac{\left[\left(r_{\mathrm{o}}+u\right)^{2}-\left(r_{\mathrm{i}}+u\right)^{2}\right]-\left[r_{\mathrm{o}}^{2}+r_{\mathrm{i}}^{2}\right]}{r_{0}^{2}-r_{\mathrm{i}}^{2}} \tag{8.2-45}$$

Eq. (8.2-45) reduces to

$$\Delta V\left(u\right)=\frac{2u}{r_{\mathrm{o}}-r_{\mathrm{i}}} \tag{8.2-46}$$

In the determination of the circumferential strain in a cracked fuel cell, it is assumed that the fuel continues to act as linear elastic material obeying Eq. (8.2-3). With the circumferential and axial stresses set to the negative of the plenum pressure, Eq. (8.2-35), and the radial stress as given in Eq. (8.2-36), the circumferential strain in the cracked zone can be determined as the sum of the force and thermal component to the strain $\varepsilon_{\theta,\mathrm{f}}^{c}$ and $\varepsilon_{\theta,\mathrm{th}}^{c}$, respectively.

$$\varepsilon_{\theta}^{c}=\varepsilon_{\theta,\mathrm{f}}^{c}+\varepsilon_{\theta,\mathrm{th}}^{c} \tag{8.2-47}$$

$$\varepsilon_{\theta,\mathrm{f}}^{c}=-\left(1-2\nu\right)\frac{\Delta P_{\mathrm{g,i}}}{E_{\mathrm{c}}}-\frac{\nu R_{\mathrm{f}}}{E_{\mathrm{c}}r}\left(\Delta P_{\mathrm{g,i}}+\Delta\sigma_{\mathrm{FC,i}}\right) \tag{8.2-48}$$

$$\varepsilon_{\theta,\mathrm{th}}^{c}=\Delta\left(\alpha T\right) \tag{8.2-49}$$

The value of the force strain used in determining the strain of a particular cell is the average of the strain at the cell boundaries and is based on the changes occurring during the time step. The thermal component is evaluated from the change in the cell temperature during the time step.

8.2.3 Fully Cracked Fuel

In some pretransient situations and many transient cases, it is possible for the solid fuel to become fully cracked, i.e., there is no solid fuel annulus. Being fully cracked, there would be no resistance to radial relocation outward, until the cladding is reached, or radially inward, until the crack volume is closed in the central solid regions of the fuel. If the pressure in the cracks remains the plenum pressure, Eq. (8.2-38) then represents the necessary equilibrium condition that must be satisfied. However, if the crack pressure is assumed to reach a level equivalent to the central cavity pressure, the substitution in Eq. (8.2-11) yields a constant pressure, equivalent to the cavity pressure, throughout the fuel and as the fuel-cladding interface pressure. Both options are available within DEFORM-4.

The movement of the cracked fuel is controlled by the two external forces applied: (1) the central cavity pressure, and (2) the fuel-cladding interface pressure. If no molten cavity exists with a pressure greater than the plenum pressure, the cracked fuel cannot relocate radially outward to remove the fuel-cladding gap that may exist. Under these conditions the maximum fuel-cladding interface pressure is equivalent to the plenum pressure. If the previous time step contained a solid fuel annulus which had produced an interface pressure greater than the plenum pressure, once full cracking is achieved the interface pressure will drop to the plenum pressure, and the fuel may be relocated inward in response to the alleviation of previous elastic strains in the cladding.

If a pressurized molten cavity does exist at the time the cracked region reaches the melt boundary, or melting proceeds to the cracked boundary, then radial relocation to the cladding surface will take place if a gap existed. If no gap existed, then the relocation could occur either outward or inward depending on the previous interface pressure, the cavity pressure, and the option used for the pressure of the gas in the cracks.

When the crack pressure is assumed to remain at the plenum pressure, the new interface pressure is determined from Eq. (8.2-38).

$$\sigma_{\text{FC}} = -P_{\text{g}} + \frac{R_{\text{cav}}}{R_{\text{f}}} \left(\sigma_{\text{cav}} + P_{\text{g}}\right) \tag{8.2-50}$$

where

σ_{cav} = Molten central cavity pressure

R_{cav} = Outer radius of the molten central cavity

If this new interface pressure is less than the previous time step, removal of the previous elastic strain will take place, moving the cracked fuel toward the center. If it is greater than the previous interface pressure, more elastic strain will be produced, and the fuel and cladding will move outward. If this new pressure produces a circumferential stress larger than the cladding flow stress, the cladding will strain plastically to provide enough volume to reduce the cavity pressure to a value that produces an interface pressure equal to that necessary to produce a circumferential stress equal to the flow stress.

These conditions are all handled by the fully crack fuel solution driver subroutine MKDRIV.

With these considerations it is possible for the cladding stress to build up while solid fuel exists, then become alleviated when the fuel becomes fully cracked with a low molten cavity pressure, and then again build up as the molten cavity pressure and melt radius increase. The timing of such behavior and the magnitude reached will depend on the particular transient being studied.

8.2.4 Axial Expansion Solution

As discussed in Section 8.2.1, the analytical solution to the mechanics equations is produced through a generalized plane strain assumption. The axial interfaces between segments are assumed to remain parallel, and a segment expands or contracts with a uniform strain, z_{o}, over its entire radius. Since this axial strain exists in the equations which represent the radial displacement function, it is necessary to find the axial strain prior to the radial strain results. The calculation for the axial strain of a segment is performed in the subroutine EXPAND.

The axial force in the fuel segment, F_{f}, is found by integrating the axial stress, σ_{z}, over the cylindrical fuel geometry,

$$F_{\text{f}} = 2\pi \int_{\rho}^{\eta} \sigma_{\text{z}} r \text{dr} \tag{8.2-51}$$

Substituting Eq. (8.2-27), Eq. (8.2-28), Eq. (8.2-30), Eq. (8.2-31) and Eq. (8.2-10) into Eq. (8.2-4) and solving for the axial stress function, yields:

$$\sigma_{\text{z}} = \frac{2\nu E}{\left(\eta^2 - \rho^2\right)} \left[\frac{I\left(\eta\right)\eta^2}{\left(1-\nu\right)} - \frac{\left(\rho^2 - \sigma_\rho - \eta^2\sigma_\eta\right)}{E}\right] - \frac{E\Delta\left(\alpha T\right)}{\left(1-\nu\right)} + Ez_O \tag{8.2-52}$$

This is then used in Eq. (8.2-51) and integrated to yield

$$F_{\text{f}} = -2\pi E\eta^2 I\left(n\right) + 2\pi\nu\left(\eta^2\sigma_\eta - \rho^2\sigma_\rho\right) + \pi E z_O\left(\eta^2 - \rho^2\right) \tag{8.2-53}$$

In order to find the axial plane strain, a total force balance is performed. The state of the fuel-cladding gap can influence the terms in the force summation. If the fuel-cladding gap is open, or the free axial expansion option is chosen through the input parameter NAXOP, then the force summation contains no term for the cladding effects. If, however, the fuel and cladding are in contact, then cladding terms must be included in the force balance. The following is the general equation for the force balance.

$$F_{\text{f}} = F_{\text{cav}} + F_{\text{ax}} + F_{\text{c}} \tag{8.2-54}$$

where

F_{cav} = Force in the central void or molten fuel cavity

F_{ax} = Force applied axially to the fuel column, usually from the plenum gas pressure

F_{f} = Force from the solid fuel zone

F_{c} = Force from the cladding, which is 0 with free axial expansion

In the central void or molten cavity, the force is given by

$$F_{\text{cav}} = \pi \rho^2 P_{\text{cav}} \tag{8.2-55}$$

where

P_{cav} = The pressure in the molten cavity or central void

ρ = The radial extent of melting or the central void radius

The axial force from the plenum pressure is given by

$$F_{\text{ax}} = \pi r_\rho^2 P_{\text{gas}} \tag{8.2-56}$$

where

P_{gas} = Fission-gas plenum pressure

r_{p} = Radius of the plenum

For the cladding, the force is either zero or the same as Eq. (8.2-53) but with cladding properties, thermal expansion, and inner and outer boundary forces.

If free axial expansion is assumed, then Eq. (8.2-53), Eq. (8.2-55), and Eq. (8.2-56) can be used in Eq. (8.2-54). This is then solved for the uniform axial strain.

$$z_{\text{o}} = \left[z_{\text{o}} \left(\text{thermal} \right) + z_{\text{o}} \left(\text{ forces } \right) \right] A_{\text{f}} \tag{8.2-57}$$

$$z_{\text{o}} \left(\text{thermal} \right) = \frac{2 \eta^2 I \left(\eta \right)}{\left(\eta^2 - \rho^2 \right)} \tag{8.2-58}$$

$$z_{\text{o}} \left(\text{forces} \right) = \frac{-2 \upsilon}{\left(\eta^2 - \rho^2 \right)} \frac{\left(\eta^2 \sigma_\eta - \rho^2 \sigma_\rho \right)}{E} - \frac{\left(\rho^2 P_{\text{cav}} + r_\rho^2 P_{\text{gas}} \right)}{E} \tag{8.2-59}$$

where

A_{f} = Fraction of axial expansion to be used

All properties in the above two equations refer to the fuel, and the modulus of elasticity, E, is a mass-averaged value.

Since this equation assumes an initial stress free state, it must be modified to account for the changes, which take place during the specific time step. The time step changes occur in the inner and outer boundary conditions, σ_ρ and ρ_η, and the cavity and plenum pressures, P_{CAV} and P_{gas}.

$$z_{\text{o}} \left(\text{forces} \right) = \frac{-2 \upsilon}{\eta^2 - \rho^2} \frac{\left(\eta^2 \Delta \sigma_{\eta,\text{i}} - \rho^2 \Delta \sigma_{\rho,\text{i}} \right)}{E} - \frac{\left(\rho^2 P_{\text{cav}} + r_\rho^2 P_{\text{gas}} \right)}{E \left(\eta^2 - \rho^2 \right)} \tag{8.2-60}$$

Eq. (8.2-60) together with Eq. (8.2-58) define the axial strain occurring during the time step due to force and thermal considerations, respectively.

For the case where the fuel and cladding are considered "constrained", the fuel and cladding surfaces are assumed to be locked to each other. Eq. (8.2-54) is used with Eq. (8.2-53) (used twice, once with fuel properties, once with cladding properties), and Eq. (8.2-55) and Eq. (8.2-56). The result is again solved for the axial strain

$$z_{\mathrm{o}}\left(\text{thermal}\right) = \frac{2\left[E_{\mathrm{f}}\eta_{\mathrm{f}}^{2} I\left(n_{\mathrm{f}}\right) + E_{\mathrm{c}}\eta_{\mathrm{c}}^{2} I\left(\eta_{\mathrm{c}}\right)\right]}{\left[E_{\mathrm{c}}\left(\eta_{\mathrm{c}}^{2} - \rho_{\mathrm{c}}^{2}\right) + E_{\mathrm{f}}\left(\eta_{\mathrm{f}}^{2} - \rho_{\mathrm{f}}^{2}\right)\right]} \tag{8.2-61}$$

$$z_{0}\left(\text{forces}\right) = \frac{-2\left[v_{\mathrm{f}}\left(\eta_{\mathrm{f}}^{2}\sigma_{\eta,\mathrm{f}} - \rho_{\mathrm{f}}^{2}\sigma_{\rho,\mathrm{f}}\right) + v_{\mathrm{c}}\left(\eta_{\mathrm{c}}^{2}\sigma_{\eta,\mathrm{c}} - \rho_{\mathrm{c}}^{2}\sigma_{\rho,\mathrm{c}}\right)\right] - \left(\rho_{\mathrm{f}}^{2}P_{\mathrm{cav}} + r_{\mathrm{p}}^{2}P_{\mathrm{gas}}\right)}{E_{\mathrm{c}}\left(\eta_{\mathrm{c}}^{2} - \rho_{\mathrm{c}}^{2}\right) + E_{\mathrm{f}}\left(\eta_{\mathrm{f}}^{2} - \rho_{\mathrm{f}}^{2}\right)} \tag{8.2-62}$$

The subscript "f" refers to fuel properties and dimensions and the subscript "c" refers to the cladding properties and dimensions. Again, Eq. (8.2-62) must be transformed to consider the changes from one time step to the next.

$$z_{0}\left(\text{forces}\right) = \frac{-2\left[v_{\mathrm{f}}\left(\eta_{\mathrm{f}}^{2}\Delta\sigma_{\eta,\mathrm{f}} - \rho_{\mathrm{f}}^{2}\Delta\sigma_{\rho,\mathrm{f}}\right) + v_{\mathrm{c}}\left(\eta_{\mathrm{c}}^{2}\Delta\sigma_{\eta,\mathrm{c}} - \rho_{\mathrm{c}}^{2}\Delta\sigma_{\rho,\mathrm{c}}\right)\right] - \left(\rho_{\mathrm{f}}^{2}\Delta P_{\mathrm{cav}} + r_{\mathrm{p}}^{2}\Delta P_{\mathrm{gas}}\right)}{E_{\mathrm{c}}\left(\eta_{\mathrm{c}}^{2} - \rho_{\mathrm{c}}^{2}\right) + E_{\mathrm{f}}\left(\eta_{\mathrm{f}}^{2} - \rho_{\mathrm{f}}^{2}\right)} \tag{8.2-63}$$

Eq. (8.2-63) together with Eq. (8.2-61) defines the axial strain for the time step when the fuel and cladding are locked together.

The axial strain is separated into its thermal and force components to allow for the option to use only thermal effects or both thermal and force effects through the input parameter IAXTHF. There are also four options available which concern the treatment of the fuel-cladding locking. The axial expansion can be set to the free axial expansion of the fuel or the cladding, to always be constrained expansion, or to be a combination depending on the actual interface conditions. In the fourth option, all axial segments below the top locked segment are assumed to be in the constrained state. These options are controlled through the input parameter NAXOP. The most realistic options are both thermal and force components with the mixed free/constrained state. The other options are available to facilitate the study of axial expansion assumptions on accident sequences.

There also may be a third fuel-cladding boundary condition, which does not currently exist in DEFORM-4. This is the "slip" condition where, at some level of mismatch between the fuel and cladding forces, the fuel would slide along the cladding. This is noted for possible inclusion in the future.

In the equations given above, the implicit assumption is present that the fuel in the region under consideration is both solid and uncracked. In the current version of DEFORM-4, the entire solid fuel zone is used in the calculation, both continuous and cracked. The result is to provide average axial expansion effects in a generalized plane strain framework. If transverse crack volumes are instituted in the cracking model, then the expansion effects in the cracked fuel region will be restudied. In order to provide a conservative assumption, when the melting reaches the cracked fuel boundary, axial expansion of that node is stopped. This is used to handle the necessity of expanding into transverse crack volume, before additional reactivity effects are treated.

8.3 Fuel-pin Phenomenology

In addition to the area of investigation termed "mechanics" that was described in Section 8.2, there is another area which is broadly termed "phenomenology". This includes those elements of fuel performance that are not necessarily considered in a structural/mechanical sense, but are generally of a microscopic nature resulting in macroscopic effects. Examples would include as-fabricated porosity migration, grain growth, fission-gas generation and release, and fuel swelling induced by solid and volatile fission products. Since these produce effects on various time scales, some are not considered in the transient calculation. However, all models have been coded so that they could be incorporated in the transient calculation if appropriate.

8.3.1 As-Fabricated Porosity Migration

When cylindrical oxide fuel pins are placed in a neutron flux, the volumetric heating rates and low thermal conductivity of the fuel combine to produce high temperatures and very steep radial thermal gradients. These conditions can lead to the phenomenon commonly referred to as restructuring. The most distinct macroscopic aspects are divided into the columnar, equiaxed, and as-fabricated fuel zones. The basic physical processes that produce these zones have been identified as grain growth kinetics for the equiaxed zone, and porosity migration for the columnar zone. In this section, the phenomenon of porosity migration will be discussed.

Sintered fuel pellets contain residual pores on the grain boundaries. At the high temperatures commonly experienced in a nuclear fuel pin, the mobility of the constituent atoms can become important since this activity is usually related to the internal energy, which is represented in an Arrhenius equation form. However, if the temperature was uniform, there would be no macroscopic movement because there is no driving force. The large thermal gradients that exist in a fuel pin act as the driving force for atomic movement. This mobility and driving force cause the pores to migrate up the thermal gradient. This movement of porosity is important because the thermal conductivity of the fuel depends on its local porosity. If there is a large amount of pore migration a central hole will be formed. This change in geometry affects the heat transfer characteristics of the pin.

The process begins with the coalescence of the irregularly shaped pores in the as-fabricated fuel. The coalescence and movement of these pores results in the formation of characteristic lenticular pores, with a long axis parallel to the fuel isotherms and the short axis in the direction of the thermal gradient. This initial step of becoming lenticular is not included in DEFORM-4; only the movement of the pores up the thermal gradient.

There are three mechanisms that could lead to pore motion: (1) evaporation condensation across the pore, (2) pore surface diffusion, and (3) mass diffusion around the pores. The mass diffusion process is assumed to be negligible because of the high activation energy required to make atoms in a solid sufficiently mobile to produce appreciable mass transport. The evaporation-condensation process is expected to be the dominant process if the as-fabricated pores are large and temperatures are high. If the pores are small and at lower temperatures, the surface diffusion process would be expected to dominate. When the LIFE-III [8-5] code was in the process of thermal calibration, it was found that the available data made it impossible to determine the thermal dependences of these last two processes, so the evaporation-condensation process was chosen as the dominant mechanism. This same approach has been employed in DEFORM-4.

The large radial thermal gradient existing in reactor fuel pins at power produces a gradient across the pore. The atoms on the hotter surface evaporate, move across the pore, and condense on the cooler surface. This causes the pore to move up the thermal gradient. Bober and Schumacher [8-6] developed the following form for the velocity of the pore due to this process.

$$U = \frac{A_{\mathrm{p}}}{T^{A}} e^{-\left(\frac{Q_{\mathrm{p}}}{RT}\right)} \frac{\mathrm{dT}}{\mathrm{dr}} \tag{8.3-1}$$

where

U = Pore velocity, m/s

T = Temperature, K

r = Radius, m

A_{p} = Pre-exponential factor, $\mathrm{m}^2\ \mathrm{T}^{(\mathrm{A}-1)}\ \mathrm{s}^{-1}$

Q_{p} = Evaporation-condensation activation energy, J g-mole^{-1}

R = Universal gas constant, J K^{-1} g-mole^{-1}

A = Temperature exponent

Theoretical values for Q_{p}, A_{p}, and A were obtained by Clement [8-7] and compare well with experimental values. Values have also been determined through the thermal calibration of the LIFE-III code [8-5].

In a cylindrical fuel rod with an axisymmetric power distribution, the thermal gradient is zero at the center of the pin, or inner surface if a central void exists. The direct application of Eq. (8.3-1) would result in an accumulation of porosity in the innermost cell. To avoid this nonphysical situation, the thermal gradient at the inner fuel cell boundary is assumed to be the average value across the central cell. This treatment simulates the diffusion of the pores and the formation of channels open to the central void.

In DEFORM-4, each fuel cell is assumed to have a uniform porosity. The change in cell porosity is determined from the initial porosity, the porosity moving into the cell from a neighboring cell, and the porosity moving out of the cell to a neighboring cell.

$$P_{\mathrm{i}}\left(t+\Delta t\right)=P_{\mathrm{i}}\left(t\right)+P_{\mathrm{in}}\left(\Delta t\right)-P_{\mathrm{out}}\left(\Delta t\right) \tag{8.3-2}$$

where

$P_{\mathrm{i}}\left(t+\Delta t\right)$ = Porosity in cell i at the end of the time step

$P_{\mathrm{i}}\left(t\right)$ = Porosity in cell i at the beginning of the time step

$P_{\mathrm{in}}\left(\Delta t\right)$ = Porosity moving into cell i from cell $i+1$

$P_{\mathrm{out}}\left(\Delta t\right)$ = Porosity moving out of cell i and entering cell $i-1$

Δt = Time-step length

Lackey, et al. [8-8] developed the amount of porosity crossing a cell boundary, based on the velocity of the pores and the length of the time step. The porosity in the annulus from r_{i} to $r_{\mathrm{i}}+U_{\mathrm{i}}\Delta t$ would be expected to cross the cell boundary at r_{i} during the time step, Δt. These considerations result in the following definitions for P_{in} and P_{out}.

$$P_{\mathrm{in}}\left(\Delta t\right)=\frac{\pi P_{\mathrm{i+1}}\left(t\right)}{A_{\mathrm{i+1}}}\left[\left(r_{\mathrm{r+1}}+U_{\mathrm{i+1}}\Delta t\right)^{2}-r_{\mathrm{i+1}}^{2}\right] \tag{8.3-3}$$

$$P_{\mathrm{out}}\left(\Delta t\right)=\frac{\pi P_{\mathrm{i}}\left(t\right)}{A_{\mathrm{i}}}\left[\left(r_{\mathrm{i}}+U_{\mathrm{i}}\Delta T\right)^{2}-r_{\mathrm{i}}^{2}\right] \tag{8.3-4}$$

where

$P_{\mathrm{i+1}}\left(t\right), P_{\mathrm{i}}\left(t\right)$ = Initial porosity in fuel cells $i+1$ and i, respectively

$r_{\mathrm{i+1}}$, r_{i} = Outer and inner radial boundaries of fuel cell i

$U_{\mathrm{i+1}}$, U_{i} = Pore velocity at radial locations $r_{\mathrm{i+1}}$ and r_{i}, respectively

$A_{\mathrm{i+1}}$, A_{i} = Cross-sectional area of radial fuel cells $i+1$ and i, respectively

The application of Eq. (8.3-2) through Eq. (8.3-4) could lead to a situation in which more porosity leaves a cell than exists originally and enters from the neighboring cell. This is clearly a nonphysical result of the equations. It is also unlikely that the porosity in any cell would become zero because porosity can become trapped behind dislocations, impurities, and fission products. A parameter, PRSMIN, is available to allow input of the minimum porosity allowed in a cell. If application of Eq. (8.3-2) would cause the new cell porosity to fall below PRSMIN, P_{out} is reduced to the amount, which would make the cell porosity equal to the allowed minimum. This treatment assumes that the total porosity is conserved.

At the fuel surface, the influx of porosity is assumed to be zero. Because of the strong temperature dependence of the Arrhenius term in Eq. (8.3-1), the pore velocity is very small even with the thermal gradient at its maximum. This, combined with the surface tension, makes it extremely unlikely to have significant porosity introduced at the fuel surface.

The above equations are solved in the subroutine PORMIG. In the solution, it is assumed that all pores in the vicinity of the boundary r_{i} travel at the velocity U_{i}. Therefore, the maximum travel of a pore is directly related

to the time-step length.

$$\Delta r_{\mathrm{v}} = U_{\mathrm{i}} \Delta t \tag{8.3-5}$$

where

Δr_{v} = Maximum distance traveled by pores

In order to maintain accuracy, it has been found that the time step length should be restricted so that the maximum distance traveled by the fastest pores is less than a fourth of the cell width.

$$\Delta t_{\mathrm{m}} = \frac{\Delta r_{\mathrm{m}}}{4 U_{\mathrm{m}}} \tag{8.3-6}$$

where

Δt_{m} = Maximum time-step length

Δr_{m} = Extent of the fuel cell outside the fastest velocity boundary

U_{m} = Maximum pore velocity

There is a second reason for limiting the size of the computational time step. As discussed in Section 8.1.2, the thermal hydraulic calculations in SAS4A are performed separately from the mechanical/phenomenological calculations in DEFORM-4. Since porosity migration changes the geometry of the fuel pin and the radial porosity distribution, which changes the conductivities, it is necessary to limit the time step so the geometric and property changes can be fed back into the thermal calculation. It has been found that the criterion given in Eq. (8.3-6) is quite adequate for this purpose.

When the porosity migrates up the thermal gradient, it can pull the grain boundaries along with it, producing the characteristic columnar grains seen in restructured fuel. If enough porosity has migrated out of a cell, the grains in the cell are assumed to be columnar. In DEFORM-4, the mechanism for determining if a cell contains columnar grains is the checking of the current porosity against the initial value in the as-fabricated fuel.

$$P_{\mathrm{i}}\left(t + \Delta t\right) \leq R_{\mathrm{eq}} P_{\mathrm{o}} \tag{8.3-7}$$

where

R_{eq} = Fraction, input parameter

P_{o} = Initial as-fabricated porosity

If the inequality in Eq. (8.3-7) is satisfied, the cell is assumed to contain columnar grains. All cells inside the outermost cell that satisfy the inequality are also assumed to be columnar. This determination does not affect any calculations, but is used for comparisons with the results from the destructive examination of pins used in calibration exercises.

The movement of these as-fabricated pores causes a movement of gas to the central void, and release to the plenum. In order to achieve a strict conservation of gases within the pin, the effect must be considered. At the as-fabricated conditions, the porosity is assumed to contain helium in equilibrium with the reference conditions. This, therefore, defines the amount of gas associated with each radial cell. As the porosity changes in the cell, the associated gas is also changed under the assumption that the gas content change is directly related to the porosity change.

$$G_{\mathrm{af,i}}\left(t + \Delta t\right) + G_{\mathrm{af,i}}\left(t\right) \frac{P_{\mathrm{i}}\left(t + \Delta t\right)}{P_{\mathrm{i}}\left(t\right)} \tag{8.3-8}$$

where

$G_{\mathrm{af,i}}\left(t + At\right)$ = As-fabricated gas in radial cell i at end of the time step, kg

$G_{\text{af,i}}(t)$ = As-fabricated gas in radial cell i at end of the time step, kg

Once the changes have been determined for all radial cells, the amount of gas released to the central void, and hence the plenum, is determined and added to the total helium inventory.

While this gas accountability may have a very small effect on the actual plenum pressure during the pretransient calculations, it can be significant under transient conditions. Since melting of a radial cell is assumed to release the as-fabricated porosity and its associated gas into the molten cavity immediately, there can arise situations where this addition can affect the molten cavity pressure. With the conservation of this material considered in DEFORM-4, a better representation of the molten cavity pressurization is provided, no matter the transient under study.

8.3.2 Grain Growth

At the relatively low temperatures in the outer fuel region, the as-fabricated porosity is unable to migrate despite the large thermal gradient, because the atomic mobility is too low. However, the atoms may be active enough to cross the grain boundaries. The larger grains grow at the expense of the smaller, due to the tendency of atoms to jump from a convex (higher energy) to a concave (lower energy) surface. The net effect is to reduce the surface area, and thereby, the surface energy associated with the grains. This grain growth is a strong function of atomic activity, i.e., temperature. In nuclear fuels, the surface temperature is usually below the "threshold" temperature where activity is great enough to cause redistribution at the grain surfaces. Due to the strong temperature dependence and the steep thermal gradient, a distinct region usually develops where the grains grow isotropically, irrespective of the large gradient. This "equiaxed" zone extends inward to the region where pore migration becomes active and produces the "columnar" grains by dragging the boundaries during migration.

The grain-size distribution is important because the fission-gas release and fuel-creep functions depend on this parameter. The calculation of this clearly visible zone also offers a simple experimental calibration region, which can be used in the validation process.

Two grain-growth models are available in the GRGROW subroutines and are selected through the input variable NGRAIN. If NGRAIN is greater than 0, the unlimited grain-growth option is used and the value of NGRAIN is the grain diameter exponent; see Section 8.3.2.1 below. If NGRAIN is zero, a limited grain-growth model is used. In this model the grain sizes are limited to an experimentally determined value; see Section 8.3.3.2 below. Both models give very similar results in the lower temperature regions associated with the equiaxed region where the grains are usually two to ten times the initial size. At higher temperatures, the first model results in larger grains because of the unconstrained growth. However, in these regions, porosity migration also produces the columnar grain structure which uses a separate method for determining the effective grain size for use in the fission-gas release and fuel-creep calculations; see Section 8.3.2.3 below. Because the unlimited growth model offers more flexibility to model mechanistic behavior, it is suggested for use prior to the validation of the integral code.

8.3.2.1 Unlimited Grain Growth - Equiaxed Region

The growth phenomenon may be characterized by a simple kinetic equation, with the growth rate given as

$$\frac{1}{D}\frac{\mathrm{dD}}{\mathrm{dt}} = \frac{1}{D^n} A_{\mathrm{g}} \exp\left(\frac{Q_{\mathrm{v}}}{RT}\right) \tag{8.3-9}$$

where

D = Grain diameter, m

n = Growth mechanism input parameter

A_{g} = Pre-exponential constant

Q_v = Activation energy for the growth mechanism related to n, J g-mole^{-1}

t = Time, s

R = Universal gas constant, J K^{-1} g-mole^{-1}

T = Temperature, K

The value of the parameter n is related to the grain-growth mechanism and depends on the driving and retarding forces being considered. Nichols [8-9] developed theoretical meanings for the values usually associated with n. If the grain boundaries are assumed to move toward their center of curvature at a rate proportional to the curvature, n would be 2. If the mechanism is through the evaporation-condensation process across pores on the boundaries, with the pressure in the pores inversely proportional to their radius, the value of n is 3. If boundaries are shifted by volume diffusion moving material around the pores or the evaporation-condensation process with the internal pore pressure constant, the value of n is 4. If the mechanism is surface or interface diffusion in the pores, the value of n is 5. In all cases where n is 3 through 5, it is assumed that the pores remain on the grain boundaries.

If the temperature is assumed constant for a time period Δt, Eq. (8.3-9) can be integrated to yield

$$D^n(t+\Delta t) = D^n(t) + nA_g \Delta t \exp\left(-\frac{Q_v}{RT}\right) \tag{8.3-10}$$

The term (nA_g) is usually combined when determining the constants by comparison with experimental data, so Eq. (8.3-10) is rewritten as

$$D^n(t+at) = D^n(t) + G_k \Delta t \exp\left(-\frac{Q_v}{RT}\right) \tag{8.3-11}$$

which is in the form coded in the GRGROW subroutine. In all cases, the values for the parameters G_k and Q_v depend strongly on the value of n used in Eq. (8.3-11).

R.N. Singh [8-10] conducted an investigation into the grain-growth kinetics for sintered UO_2 pellets at temperatures between 1800 and 2100°C. His conclusions suggested that the cubic form of Eq. (8.3-10) was most accurate, and appropriate constants were determined. When the MATPRO-10 [8-11] materials properties package was developed, the available experimental evidence was collected, and curve fits were applied for exponents of 2 through 4. In this study, it was found that an exponent of 4 gave the best fit with 3 giving very similar results. In the GRGROW subroutine, the input variables for the exponent, pre-exponential constant, and activation energy are used to provide maximum flexibility for the user.

8.3.2.2 Limited Grain Growth - Equiaxed Region

In a study of grain sizes in irradiated fuel by Ainscough, et al. [8-12], a kinetics equation for grain growth was developed that included a maximum grain size. As the grains grow, the boundaries are retarded by the effects of intergranular pores, solid fission products, and gas bubbles. After the grains reach a certain size, they are stopped from additional growth. It was postulated that the maximum grain size could be represented by the form

$$D_m = G_m \exp\left(\frac{Q_m}{RT}\right) \tag{8.3-12}$$

where

D_m = Maximum grain size, m

G_m = Pre-exponential constant, m

Q_m = Maximum grain size activation energy, J (g-mole)$^{-1}$

R = Universal gas constant, J k^{-1} g-mole^{-1}

T = Temperature, K

The temperature dependence of Eq. (8.3-12) results from the higher mobility of the retardants and the resultant reduction in grain-boundary drag as the temperature increases.

The kinetic equation developed is then given by

$$\frac{\mathrm{dD}}{\mathrm{dt}} = G\left(\frac{1}{D} - \frac{1}{D_{\mathrm{m}}}\right)\exp\left(\frac{Q}{RT}\right) \tag{8.3-13}$$

where

D = Grain diameter, m

G = Pre-exponential grain growth constant, $\mathrm{m}^2\ \mathrm{s}^{-1}$

Q = Grain-growth activation energy, J g-mole^{-1}

If the temperature is assumed to be constant over the time step, Δt, the integration of Eq. (8.3-13) produces the following transcendental equation:

$$\begin{aligned} f\left(D\left(t+\Delta t\right)\right) = D_{\mathrm{m}}^2 \ln\left(\frac{D_{\mathrm{m}} - D\left(t\right)}{D_{\mathrm{m}} - D\left(t+\Delta t\right)}\right) - G\Delta t\left(\frac{-Q}{RT}\right) \\ + D_{\mathrm{m}}\left(D\left(t\right) - D\left(t+\Delta t\right)\right) = 0 \end{aligned} \tag{8.3-14}$$

The solution for $D\left(t+\Delta t\right)$ is obtained through the Newton's Method iterative scheme.

$$D_{\mathrm{k+1}} = D_{\mathrm{k}} - \frac{f\left(D_{\mathrm{k}}\left(t+\Delta t\right)\right)}{f'\left(D_{\mathrm{k}}\left(t+\Delta t\right)\right)} \tag{8.3-15}$$

where

D_{k} = The k-th estimate of the grain diameter to satisfy Eq. (8.3-14), m

$$f'\left(D_{\mathrm{k}}\left(t+\Delta t\right)\right) = \frac{\mathrm{df}\left(D_{\mathrm{k}}\right)}{\mathrm{dD_k}} \tag{8.3-16}$$

Since the grain size at the beginning of the time step, $D\left(t\right)$, and the maximum grain size, D_{m}, are known constants, the differentiation in Eq. (8.3-16) may be performed on Eq. (8.3-14) after expanding the log term to the difference of two 1og terms:

$$f'\left(D_{\mathrm{k}}\left(t+\Delta t\right)\right) = \frac{D_{\mathrm{m}}^2}{D_{\mathrm{m}} - D_{\mathrm{k}}\left(t+\Delta t\right)} - D_{\mathrm{m}} \tag{8.3-17}$$

Eq. (8.3-17) is reduced to

$$f'\left(D_{\mathrm{k}}\left(t+\Delta t\right)\right) = \frac{D_{\mathrm{m}} D_{\mathrm{k}}\left(t+\Delta t\right)}{D_{\mathrm{m}} - D_{\mathrm{k}}\left(t+\Delta t\right)} \tag{8.3-18}$$

and this is substituted into Eq. (8.3-15) to produce the final form for the next estimate, $k+1$, of the grain size to satisfy Eq. (8.3-14).

$$D_{\mathrm{k+1}}\left(t+\Delta t\right) = D_{\mathrm{k}}\left(t+\Delta t\right) - f\left(D_{\mathrm{k}}\left(t+\Delta t\right)\right)\frac{\left(D_{\mathrm{m}} - D_{\mathrm{k}}\left(t+\Delta t\right)\right)}{D_{\mathrm{m}} D_{\mathrm{k}}\left(t+\Delta t\right)} \tag{8.3-19}$$

This iteration is continued until a consistent value is determined.

If the maximum grain size, D_{m}, is smaller than the size at the beginning of the time step, $D\left(t\right)$, due to power or temperature reductions, the current grain size, $D\left(t+\Delta t\right)$, is maintained at its previous value.

8.3.2.3 Columnar Grain Size and Region Boundaries

The extent of the columnar region boundary is found as was discussed in Section 8.3.1. The effective grain size in this region is determined by the equiaxed grain size at the columnar/equiaxed boundary and the extent of the columnar boundary.

$$D_{\text{col}} = \sqrt{\Delta r_{\text{col}} D_{\text{b}}} \tag{8.3-20}$$

where

D_{co1} = Effective grain size in the columnar region, m

Δr_{co1} = Radial extent of the columnar region, m

D_{b} = Equiaxed grain size at the columnar/equiaxed boundary, m

The boundary between the equiaxed and as-fabricated region is based on the amount of grain growth. The grain size in each radial cell is compared to the initial grain size and if suitable growth has occurred, the region is classified as equiaxed.

$$D_{\text{i}}\left(t + \Delta t\right) \geq R_{\text{ueq}} D_O \tag{8.3-21}$$

where

$D_{\text{i}}\left(t + At\right)$ = Grain size in radial cell i at the end of the time step, m

R_{ueq} = Input factor

D_{O} = As-fabricated grain size, m

The check is started at the outer fuel surface and once the inequality in Eq. (8.3-21) is satisfied, that determines the equiaxed/as-fabricated boundary.

8.3.3 Fission-gas Release

The nuclear fission processes occurring in the fuel during the irradiation produce both solid and gaseous fission products. The gaseous products are primarily xenon and krypton. The model currently used in DEFORM-4 assumes that the gaseous products either precipitate as gas-filled bubbles on the grain boundaries, are contained in microbubbles within the fuel matrix, or are released to the available free volume in the pin plenum and fuel central void. Formation of grain boundary bubbles leads to fuel swelling and reduces the fuel-cladding gap size. The intra-granular gas is assumed to play no part in fuel swelling but becomes important upon fuel melting. Release to the free volume changes the gas mixture and reduces the thermal conductivity of the gas in the gap. Fission-gas release, fuel swelling, and the fuel-pin temperature distribution are therefore closely interrelated. The migration of fission-gas bubbles up the thermal gradient is not treated.

Fission gases can be released when they reach any open porosity such as cracks, the fuel-cladding gap, the central void, or the fission-gas plenum. At temperatures below about 1300 K, the mobility of the gas atoms is too low for diffusion, so they are released only by collisions with fission fragments near the fuel surface. This fraction is very small and can be neglected. At temperatures between about 1300 and 1900 K, the atomic motion is high enough to allow diffusion to the grain surfaces. At temperatures above 1900 K, the gas bubbles become mobile and may escape by migration up the temperature gradient.

The mechanistic approach to the problem of gas release has been employed in codes such as GRASS-SST [8-13] and FRAS [8-14]. A complete modeling of the fission-gas behavior is attempted in these codes, describing the migration and coalescence of fission-gas bubbles in the grain and on grain boundaries. Gas release from the grains and grain boundaries to the exterior of the fuel are modeled. Detailed bubble-size distributions are calculated

and grain-boundary channel formation is treated. The parameters involved have been studied and extensively calibrated.

Such a complete and detailed modeling effort is not currently envisioned for SAS4A. Because of the requirements that the SAS4A code size and running time be minimized, considerably simpler, less-mechanistic models have been incorporated into SAS4A. These models relate the release rate to escape probabilities. These probabilities are modeled as functions of temperature and density. They should also be related to grain size, but are not in the current version. The calculations are performed in the subroutine RELGAS.

8.3.3.1 Fission-gas Generation

The total amount of fission gas generated in a fuel cell is related to the power of that cell. As the fission process proceeds, a number of isotopes result. Some of these fission products are volatile gases, which may coalesce to form fission-gas bubbles. It is assumed that each fission produces a constant fraction of fission-gas atoms.

$$G_{\mathrm{a}} = F \times f_{\mathrm{g}} \tag{8.3-22}$$

where

G_{a} = Number of gas atoms generated

F = Number of fissions during the time step

f_{g} = Fractional gas atoms generated per fission

While this model is not true of a single fission event, it does accurately represent the macroscopic results of a large number of fissions.

The number of fissions in an axial segment is related to the power generated.

$$F = \frac{P_{\mathrm{j}} \Delta t}{1.603 \times 10^{-13} E_{\mathrm{f}}} \tag{8.3-23}$$

where

P_{j} = Power generated by axial fuel segment j, w

Δt = Time-step length, s

E_{f} = Energy generated per fission, MeV

The numeric constant in Eq. (8.3-23) is the conversion factor from MeV to W-s. The code uses the amount of gas in units of mass, so Eq. (8.3-22) is rewritten as

$$G_{\mathrm{m}} = \frac{F \times f_{\mathrm{g}} \times MW_{\mathrm{fg}} \times 10^{-3}}{N_{\mathrm{a}}} \tag{8.3-24}$$

where

G_{m} = Mass of fission gas, kg

MW_{fg} = Molecular weight of the fission gas, amu

N_{a} = Avogadro number, 0.6025×10^{24} atoms/g-mole

The numeric constant in Eq. (8.3-24) converts g to kg. Combining Eq. (8.3-24) with Eq. (8.3-23) results in the mass of fission gas generated in the axial segment during the time step.

$$G_{\mathrm{m}} = \frac{P_{\mathrm{i}} \Delta t f_{\mathrm{g}} MW_{\mathrm{fg}}}{9.658 \times 10^{13} E_{\mathrm{f}}} \tag{8.3-25}$$

If the radial power shape was flat, the total fission-gas mass could be divided between the radial fuel cells on the basis of cell mass. However, it is possible to have a radial power shape. The fission-gas mass in a radial fuel cell is determined by multiplying the fission gas generated by the whole segment by a radial factor.

$$G_{\mathrm{i}} = G_{\mathrm{m}} f_{\mathrm{i}} \tag{8.3-26}$$

where

G_{i} = Fission-gas mass generated in radial fuel cell i, kg

f_{i} = Fraction of total mass generated in radial cell i

The radial factor is based on the radial power shape factors and the radial fuel cell masses.

$$f_{\mathrm{i}} = \frac{S_{\mathrm{i}} M_{\mathrm{i}}}{\sum_{\mathrm{k}=1}^{N} \left(S_{\mathrm{k}} M_{\mathrm{k}}\right)} \tag{8.3-27}$$

where

S_{i} = Radial power shape factor for radial cell i

M_{i} = Fuel mass in radial cell i, kg

N = Number of radial fuel cells

Substituting Eq. (8.3-27) and Eq. (8.3-25) into Eq. (8.3-26) results in the fission-gas mass generated in the radial cell.

$$G_{\mathrm{i}} = \frac{P_{\mathrm{j}} \Delta t f_{\mathrm{g}} M W_{\mathrm{fg}} S_{\mathrm{i}} M_{\mathrm{i}}}{9.658 \times 10^{13} E_{\mathrm{f}} \sum_{\mathrm{k}=1}^{N} \left(S_{\mathrm{k}} M_{\mathrm{k}}\right)} \tag{8.3-28}$$

The amount of this gas located on the grain boundaries vs the amount retained in the fuel matrix is controlled through the input parameter FIFNGB. At each time step the new gas generated is divided between these two 1ocations

$$G_{\mathrm{gb,i}} = G_{\mathrm{i}} f_{\mathrm{gb}} \tag{8.3-29}$$

$$G_{\mathrm{fm,i}} = G_{\mathrm{i}} \left(1 - f_{\mathrm{gb}}\right) \tag{8.3-30}$$

where

$G_{\mathrm{gb,i}}$ = Mass of generated fission gas assumed to be on grain boundaries in radial cell i, kg

$G_{\mathrm{fm,i}}$ = Mass of generated fission gas assumed to be in the fuel matrix in radial cell i, kg

f_{gb} = Fraction of fission gas generated that is on the grain boundaries, input parameter FIFNGB

Currently, release from these two regions is treated in the same manner, as described below. However, with this type of split in DEFORM-4, it would be possible to develop different release mechanisms, and even provide for a radially dependent value of f_{gb}. These considerations have been identified for possible future work if their effects were to become important.

This split in the gas location is significant because of the interactions with two other models: (1) the molten cavity pressure (Section 8.3.7), and (2) fission gas swelling (Section 8.3.4). Upon melting, all grain boundary gas is released immediately while the intra-granular gas has a delayed coalescence. In fuel swelling, only the grain boundary gas is assumed to produce fuel swelling.

8.3.3.2 Isotropic Fission-gas Release

In the isotropic fission-gas release model, the release is treated as a function of a single release rate fraction, f, which is a function of temperature. This is the probability per unit time that a retained gas atom would be released. The basic rate equation governing the amount of retained gas in the fuel, at all locations, can be given by

$$\frac{\mathrm{dS}}{\mathrm{dt}} = -Sf + G \tag{8.3-31}$$

where

S = Amount of fission-gas retained in the fuel, kg

f = Fractional release rate of the retained gas, kg (kg-s) $^{-1}$

G = Fission-gas production rate, kg s^{-1}

Assuming that f and G are constant within the time step, Δt, Eq. (8.3-31) may be integrated over the time step to yield the amount of retained gas at the end of the interval.

$$S\left(t+\Delta t\right) = S\left(t\right)\exp\left(-f\Delta t\right) + \frac{G}{f}\left[1-\exp\left(-f\Delta t\right)\right] \tag{8.3-32}$$

Since the mobility of the gas atoms is a thermally activated process, it is assumed that the fractional release rate can be represented by

$$f = A_\alpha \exp\left(\frac{-Q_\alpha}{RT}\right) \tag{8.3-33}$$

where

A_α = Pre-exponential input constant

Q_α = Activation energy for release, input constant

The constants A_α and Q_α are determined through comparisons with experimentally determined values of the retained fission-gas from the destructive examination of irradiated fuel pellets and through comparisons with more sophisticated fission-gas release codes. Preliminary constants have been determined, but the validation exercises will be used to refine them.

8.3.3.3 Fission-gas Trap-release Model

A slightly more complex model that includes a more mechanistic treatment of fission-gas release was developed by Weisman et al. [8-15]. In this model the gas is assumed to be released in two ways: (1) direct release to the fuel surface, and (2) entrapment in the fuel matrix and subsequent release to the surface.

The amount of gas generated that is released directly is given by the following:

$$\mathrm{dn}_1 = k'G\mathrm{dt} \tag{8.3-34}$$

where

dn_1 = Amount of gas released in time increment dt, kg

k' = Fraction of the free gas that escapes to the surface without becoming trapped

G = Fission-gas production rate, kg s^{-1}

dt = Time increment, s

The amount of gas trapped in the fuel matrix is given by the total amount generated minus the total released.

$$c = Gt - n \tag{8.3-35}$$

where

c = Amount of gas trapped in the fuel matrix, kg

t = Time, s

n = Amount of gas released, kg

If k is the probability that trapped gas will be released per unit time, then the amount of trapped gas that is released is given by

$$\mathrm{dn_f} = kc\mathrm{dt} \tag{8.3-36}$$

where

$\mathrm{dn_f}$ = Amount of trapped gas which is freed, kg

Of this gas, only a fraction k' is released to the surface without becoming retrapped

$$\mathrm{dn_2} = k'kc\mathrm{dt} \tag{8.3-37}$$

where

$\mathrm{dn_2}$ = Amount of trapped gas that is released to the fuel surface

The total amount of released gas, dn, is therefore given by

$$\mathrm{dn} = \mathrm{dn_1} + \mathrm{dn_2} \tag{8.3-38}$$

or upon substitution of Eq. (8.3-34), Eq. (8.3-35), and Eq. (8.3-37) into Eq. (8.3-38),

$$\mathrm{dn} = k'G\mathrm{dt} + k'k\left(\mathrm{Gt} - n\right)\mathrm{dt} \tag{8.3-39}$$

Integration of Eq. (8.3-39) is performed to give

$$n = G\left\{t - \frac{1-k'}{K}\left[1 - \exp\left(-Kt\right)\right]\right\} \tag{8.3-40}$$

where

$K = k'k$ = Probability that trapped gas is released from the fuel matrix to the fuel surface

Assuming that the reactor power history is described by a series of constant power steps, the amount of gas released during a constant power time step is given by

$$\begin{aligned} \Delta n_{\mathrm{i}} &= n_{\mathrm{i}} - n_{\mathrm{i}-1} \\ &= G_{\mathrm{i}}\left\{\Delta t_{\mathrm{i}} - \frac{1-k'}{K_{\mathrm{i}}}\left[1 - \exp\left(-K_{\mathrm{i}}\Delta t_{\mathrm{i}}\right)\right]\right\} \\ &\quad + c_{\mathrm{i}-1}\left[1 - \exp\left(-K_{\mathrm{i}}\Delta t_{\mathrm{i}}\right)\right] \end{aligned} \tag{8.3-41}$$

where

Δn_{i} = Amount of gas released during time step Δt_{i}, kg

G_{i} = Fission-gas generation rate during time step Δt_{i}, kg s^{-1}

Δt_{i} = Time-step duration, s

$c_{\mathrm{i}-1}$ = Amount of trapped gas at beginning of time step, kg

$k'_{\mathrm{i}}, K_{\mathrm{i}}$ = Defined above, but evaluated for the time step Δt_{i}

Equations for the terms k' and K were developed during the calibration of the FRAP-S2 computer code [8-16].

$$k' = \exp\left(-\frac{Q_{\mathrm{A1}}}{T} + Q_{\mathrm{A2}} - Q_{\mathrm{A3}}d\right) \tag{8.3-42}$$

$$K = \exp\left(-\frac{Q_{\mathrm{A4}}}{T} + Q_{\mathrm{A5}}\right) \tag{8.3-43}$$

where

T = Temperature, K

d = Percent theoretical density of the fuel

$Q_{\mathrm{A1}}, Q_{\mathrm{A2}}, Q_{\mathrm{A3}}, Q_{\mathrm{A4}}, Q_{\mathrm{A5}}$ = Input constants

8.3.4 Fuel Swelling

In DEFORM-4, the as-fabricated porosity and the fission-gas-generated porosity are treated separately. The migration of the as-fabricated porosity can lead to either densification or swelling of the fuel depending on local conditions (see Section 8.3.1). The newly formed porosity arising from fission-gas bubbles introduces additional porosity that may be in a nonequilibrium condition, depending on the amount of gas in the bubbles, the local hydrostatic pressure, and the fuel surface tension. This fission-gas porosity may increase or decrease as a function of time, producing changes in the fuel dimensions through swelling or densification. These changes in fuel porosity also affect the fuel thermal conductivity, since both the as-fabricated and fission-gas porosity are considered in the porosity terms. In addition to this gas effect, the solid fission products locate themselves interstitially in the fuel matrix, causing strains that produce swelling. Both these effects are accounted for by DEFORM-4.

Swelling strains may occur both axially and radially. In general, the axial swelling strains in oxide fuels are relatively small compared to the thermal expansion effects. However, in some transients the differences caused by including the axial swelling can be enough to modify the accident scenario. For this reason, and to provide the basis for future versatility, axial swelling has been incorporated into DEFORM-4. Radial swelling is important because of the effects on fuel-cladding gap size and mechanical interaction. Both effects can produce large differences in the prediction of cladding failure, so radial swelling is also included in DEFORM-4. These fuel swelling considerations are treated in the subroutine FSWELL.

8.3.4.1 Nonequilibrium Fission-gas Bubbles

The swelling rate due to fission gas depends on the release of the gas to grain boundaries and formation of fission-gas bubbles. Detailed treatments for this process can be found in codes such as FRAS [8-14] and GRASS-SST [8-13]. The current model in DEFORM-4 is much simpler and phenomenological on a more macroscopic level. While fission gas exists in both the fuel matrix and on grain boundaries, it is the bubbles on the grain boundaries that produce the significant swelling in oxide fuels. If these bubbles are underpressurized, a reduction in the bubble volume due to the fuel hydrostatic pressure will reduce the volume of a fuel cell. If the bubbles are overpressurized, an increase in bubble size, and thereby fuel cell volume can result. To determine the rate of swelling, or densification, the mechanical stresses (Section 8.2), internal gas pressure, pressure due to surface tension, and the creep properties of the fuel (Section 8.7.6) must be known.

While fission gas exists in both the fuel matrix and on the grain boundaries, it is the bubbles on the grain boundaries that produce significant swelling in oxide fuels. However, the actual amount of gas involved and its distribution changes with burnup. As discussed in Section 8.3.3.1, DEFORM-4 uses a fixed factor to distribute the generated fission gas between the fuel matrix and grain boundaries. However, this type of approach does not fully represent the gas mass associated with fuel swelling. Therefore, DEFORM-4 uses a burnup dependent parameter. FGMIN, to specify the amount of fuel matrix retained gas to associate with the fission gas induced swelling. Figure 8.3.1 shows the recommended curve for this fission gas parameter.

The treatment of the bubble gas pressure and surface tension follows that found in the LIFE code [8-7]. This approach is macroscopic in nature and the constants used are based on the calibration of LIFE. The swelling rate of these bubbles is estimated from the fuel creep function. Swelling causes changes in the stress state of the fuel because the changes in geometry produce changes in the boundary conditions. But changes in stress states also produce changes in the swelling through changes in hydrostatic pressures. The swelling and mechanical responses are closely coupled. For this reason, the swelling calculation in DEFORM-4 has been incorporated within the iterations to find the set of conditions that bring about consistency between the fuel and cladding. Swelling and mechanical strains are stored separately, but calculated considering mutual influences. The strains due to swelling/hot pressing of the fuel are added to the total mechanical deformation at the end of each time step.

The pressure in the bubble necessary to balance the surface tension is parameterized as

$$P_\gamma = A_{\text{pg}} e^{(Q_{\text{pg}}/\text{RT})} \tag{8.3-44}$$

where

P_γ = Pressure due to surface tension effects, Pa

A_{pg} = Pre-exponential calibration constant, Pa

Q_{pg} = Exponential calibration constant, J(g-mole)$^{-1}$

T = Temperature, K

R = Ideal-gas constant, J k^{-1}(g-mole)$^{-1}$

If the bubbles are assumed to be spherical, the relationship between surface tension pressure and bubble radius can be determined.

$$r_{\text{B}} = \frac{2\gamma}{P_\gamma} \tag{8.3-45}$$

where

r_{B} = Average fission-gas bubble radius, m

γ = Surface tension, N/m

The internal fission-gas bubble pressure, P_{fg}, is determined by the ideal-gas law.

$$P_{\text{fg}} = M_{\text{fg}} \frac{RT}{V_{\text{fg}}} \tag{8.3-46}$$

where

P_{fg} = Pressure inside the fission gas-bubble, Pa

M_{fg} = Moles of fission gas in the bubbles

R = Ideal-gas constant

T = Temperature, K

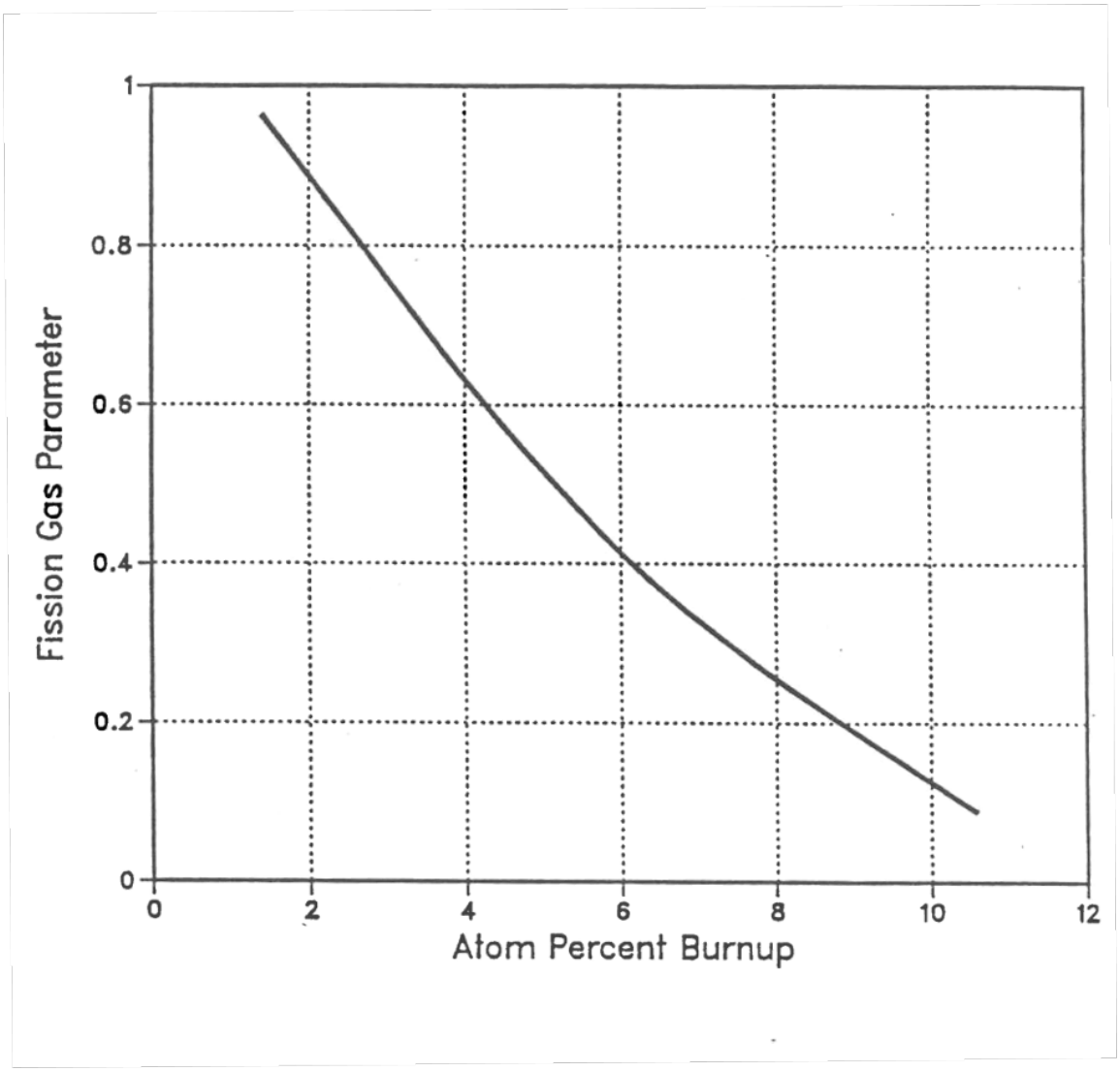


Figure 8.3.1: DEFORM-4 Fission Gas Parameter FGMIN

V_{fg} = Volume of fission-gas bubbles in the cell, m^3

The hydrostatic pressure of the fuel, P_σ, is determined from the stress state in the fuel cell. The input parameter IPSIG determines the assumption used to define this pressure according to the following table.

IPSIG	Definition of P_σ
1	$-\sigma_r$
2	$-1/2(\sigma_r + \sigma_\theta)$
3	$-1/3(\sigma_r + \sigma_\theta + \sigma_z)$

The imbalance between the bubble pressure and the external pressures is used to determine the effective creep rate of the fuel for swelling effects.

$$\Delta P = P_{fg} - P_\gamma - P_\sigma \tag{8.3-47}$$

where

ΔP = Pressure differential

P_{fg} = Pressure in the bubbles

P_γ = Surface tension pressure

P_σ = Hydrostatic pressure

If the differential is positive, the bubbles will expand, swelling the fuel cell. If it is negative, the bubbles will contract, densifying the fuel cell. The process assumed to control the rate of these volume changes is the creep properties of the fuel in the cell.

There is a bubble volume that would cause Eq. (8.3-47) to become zero. The equilibrium volume, V_e, is found by assuming the pressure differential is zero. To achieve this, the bubble pressure must balance the surface tension and hydrostatic pressures.

$$P_{fg} = P_\gamma + P_\sigma \tag{8.3-48}$$

Eq. (8.3-46) is substituted into Eq. (8.3-48) and the result rearranged to determine the fission-gas bubble equilibrium volume, V_o

$$V_c = \frac{M_{fg}RT}{(P_\gamma + P_\sigma)} \tag{8.3-49}$$

If all volume changes took place instantaneously, this would be the fission-gas bubble volume. However, it is assumed that the fuel creep properties define the rate of change of the volume and the driving force is the pressure differential. The following equation is therefore assumed to define the bubble volume rate of change.

$$\frac{dV_g}{dt} = \frac{(V_c - V_g)}{\tau_c} \tag{8.3-50}$$

where

V_g = Fission-gas bubble volume, m^3

V_e = Equilibrium fission-gas bubble volume, m^3

τ_c = Fuel creep time constant, s

t = Time, s

The fuel creep time constant, τ_c, is defined as the inverse of the fuel creep rate in the fuel cell under consideration with the pressure differential as the driving force. Integrating Eq. (8.3-50) over the time step, Δt, and rearranging to find the fission-gas bubble volume at the end of the time step results in the following.

$$\Delta V_g(t+\Delta t) = V_e\left[1-e^{-\frac{\Delta t}{\tau_c}}\right] + V_g(t)\, e^{-\frac{\Delta t}{\tau_c}} \tag{8.3-51}$$

The change in volume over the time step, $\Delta V_s(\Delta t)$, is found by subtracting the volume at the beginning of the time step, $V_g(t)$ from both sides of Eq. (8.3-51).

$$\Delta V_s(\Delta t) = [V_e - V_g(t)]\left[1-e^{-\frac{\Delta t}{\tau_c}}\right] \tag{8.3-52}$$

Eq. (8.3-52) gives the total volume change for a particular radial cell during the time step. When this change in volume takes place, part of it is radial volume expansion, typically 2/3, and the rest takes place axially. The radial component is handled nominally through the redefinition of the cell boundaries. The axial component requires special consideration because of the assumption of generalized plane strain. Each radial cell will produce a different axial strain due to the swelling process. A plane strain is determined that moves the same mass of fuel across the original axial segment boundary.

$$\Delta V_{s,i} = f_a \Delta V_s \tag{8.3-53}$$

where

$\Delta V_{s,i}$ = Volume change due to swelling for radial node i, m^3.

f_a = Fraction of total swelling taking place axially.

This change would result in a new axial segment length.

$$\Delta V_{s,i} = \Delta h_{s,i}\pi\left(r_{i+1}^2 - r_i^2\right) \tag{8.3-54}$$

where

$\Delta h_{s,i}$ = Axial segment length change from swelling, m

r_{i+1} = Outer radius of radial cell i after swelling is included, m

r_i = Inner radius of radial cell i after swelling is included, m.

Combining Eq. (8.3-53) and Eq. (8.3-54), the segment length change for radial cell i can be found

$$\Delta h_{s,i} = \frac{f_a \Delta V_s}{\pi\left(r_{i+1}^2 - r_i^2\right)} \tag{8.3-55}$$

The axial strain for the cell can then be found

$$Z_{s,i} = \frac{\Delta h_{s,i}}{h} \tag{8.3-56}$$

where

h = Axial segment length at beginning of time step, m

The fraction of the fuel mass that has moved from the original segment length, h, into the additional length, $\Delta h_{s,i}$, is the ratio of the length change to the new length, assuming uniform mass distribution within the segment

$$F_i = \frac{\Delta h_{s,i}}{h + \Delta h_{s,i}} \tag{8.3-57}$$

where

F_{i} = Fraction of original cell mass moved into length $\Delta h_{\mathrm{s,i}}$.

Using Eq. (8.3-56) and Eq. (8.3-57), this mass fraction can be defined in terms of the axial strain

$$F_{\mathrm{i}} = \frac{z_{\mathrm{s,i}}}{1 + z_{\mathrm{s,i}}} \tag{8.3-58}$$

The total mass movement across the original boundary can therefore be determined

$$M_{\mathrm{aT}} = \sum_{\mathrm{i}=1}^{N_{\mathrm{t}}} F_{\mathrm{i}} M_{\mathrm{i}} \tag{8.3-59}$$

where

M_{aT} = Total fuel mass moved out of original segment length, kg

M_{i} = Fuel mass in radial cell i

N_{t} = Total number of radial cells

A generalized axial swelling strain using Eq. (8.3-58) and Eq. (8.3-59) can now be defined that gives this same mass transfer.

$$M_{\mathrm{aT}} = \frac{z_{\mathrm{s,a}}}{1 + z_{\mathrm{S,A}}} M_{\mathrm{T}} \tag{8.3-60}$$

where

$z_{\mathrm{s,a}}$ = Generalized axial swelling strain

M_{T} = Total mass in the axial segment

Solving Eq. (8.3-60) yields the desired result of an axial strain that is uniform over the radial cross section, but produces the same mass movement as the sum of the individual radial cell components

$$z_{\mathrm{s,a}} = \frac{M_{\mathrm{aT}}}{M_{\mathrm{T}} - M_{\mathrm{aT}}} \tag{8.3-61}$$

These fission gas bubble swelling considerations are handled by the subroutine FSWELL.

8.3.4.2 Solid Fission-product Swelling

In addition to the gaseous fission-product swelling, there is solid fission-product swelling from products such as zirconium, niobium, molybdenum, the rare earths, yttrium, etc. A detailed treatment of these solid fission products, which considered their physical and chemical state to determine the partial volumes, would yield the lattice strain created. This, together with the isotopic yields from fission, would result in a mechanistic estimate of solid product swelling. This type of treatment requires more computational resources than are warranted for the magnitude of the phenomenon.

A simpler model is assumed which relates the fractional volume change to the fuel burnup.

$$\left(\frac{\Delta V}{V}\right)_{\mathrm{SP}} = \dot{\varepsilon}_{\mathrm{sfp}} B \tag{8.3-62}$$

where

$\left(\frac{\Delta V}{V}\right)_{\mathrm{SP}}$ = Fractional volume change due to solid fission products

$\dot{\varepsilon}_{\mathrm{sfp}}$ = Solid fission-product swelling rate parameter, $(\Delta V/V)$ (atom % burnup)$^{-1}$

B = Fuel burnup, atom %

Due to uncertainties in the thermodynamic state and migration characteristics of the products, the value of $\dot{\varepsilon}_{\text{sfp}}$ is specified by an input parameter.

The volume changes from solid product changes are included with those of the volatile products when determining the changes in fuel volume that may change the fuel-cladding interface conditions. However, this volume change is not affected by the local hydrostatic pressure and therefore remains a constant through the iterations mentioned in the previous section and discussed at length in Section 8.5.

8.3.5 Irradiation-induced Cladding Swelling

When 20% cold-worked 316 stainless steel material is irradiated in a fast neutron flux, there is a temperature- and flux-dependent reduction in density through the formation of irradiation-induced voids. The model in DEFORM-4 is based on an empirical correlation corresponding to stress-free swelling in the Nuclear Systems Material Handbook [8-17]. The volume change is represented by

$$\frac{\Delta V}{V} = \frac{\sum}{1-\sum} \tag{8.3-63}$$

where

$\sum$ = Negative fractional density change

$$\begin{aligned} \frac{\Delta V}{V} &= \frac{\sum}{1-\sum} \\ &= \frac{\rho_{\text{f}} - \rho_{\text{o}}}{\rho_{\text{o}}} \end{aligned} \tag{8.3-64}$$

ρ_{f} = Final immersion density

ρ_{o} = Initial immersion density

To find the change during the time step, the derivative of Eq. (8.3-63) is

$$\frac{\text{d}\left(\frac{\Delta V}{V}\right)}{\text{dt}} = \frac{1}{\left(1-\sum\right)^2}\left(\frac{\text{d}\sum}{\text{dt}}\right) \tag{8.3-65}$$

The correlation represents two characteristics of irradiation-induced swelling based on experimental findings. First, the rate of swelling for 20% cold-worked 316 stainless steel is temperature sensitive. Second, there appears to be an incubation period during which little change in density is observed. The form given in the NSMH for the negative fractional density change is

$$\sum = R\left(T\right)\left\{\phi t + \frac{1}{\alpha}\ln\left[\frac{1+\exp\left[\alpha\left(\tau-\phi t\right)\right]}{1+\exp\left(\alpha\tau\right)}\right]\right\} \tag{8.3-66}$$

where

$R\left(T\right)$ = Temperature-dependent rate parameter

ϕ = Fast neutron flux

t = Time

α = Calibration constant

τ = Incubation parameter

The derivative of Eq. (8.3-66) gives

$$\frac{\text{d}\sum}{\text{dt}} = R\left(t\right)\phi\left[1-\frac{1}{1+\exp\left[-\alpha\left(\tau-\phi t\right)\right]}\right] \tag{8.3-67}$$

The fractional volume change is, therefore, determined from

$$\Delta\left(\frac{\Delta V}{V}\right)=\frac{\mathrm{d}\left(\frac{\Delta V}{V}\right)}{\mathrm{dt}}\Delta t=\frac{\Delta t}{\left(1-\sum\right)^{2}}\frac{\mathrm{d}\sum}{\mathrm{dt}} \tag{8.3-68}$$

The swelling is assumed to be isotropic and always outward. The new cross sectional area, A, is calculated assuming 1/3 of the volume change takes place axially.

$$A\left(t+\Delta t\right)=\pi\left[r_{\mathrm{o}}^{2}\left(t+\Delta t\right)-r_{\mathrm{i}}^{2}\left(t+\Delta t\right)\right]=\pi\left[r_{\mathrm{o}}^{2}\left(t\right)-r_{\mathrm{i}}^{2}\left(t\right)\right]\left[1+\frac{2}{3}\left(\frac{\Delta V}{V}\right)\right] \tag{8.3-69}$$

where

A = Cross-sectional area

$r_{\mathrm{o}}\left(t\right)$, $r_{\mathrm{o}}\left(t+\Delta t\right)$ = Outer radius at the beginning and end of the time step

$r_{\mathrm{i}}\left(t\right)$, $r_{\mathrm{i}}\left(t+\Delta t\right)$ = Inner radius of the beginning and end of the time step

The new inner radius is calculated from

$$r_{\mathrm{i}}\left(t+\Delta t\right)=r_{\mathrm{i}}\left(t\right)\left[1+\frac{1}{3}\left(\frac{\Delta V}{V}\right)\right] \tag{8.3-70}$$

The forms given for the rate and incubation parameter are given below.

$$R\left(T\right)=0.01\left[\exp\left(0.0419+1.498\beta+0.122\beta^{2}-0.332\beta^{3}-0.441\beta^{4}\right)\right] \tag{8.3-71}$$

$$\tau=4.742-0.2326\beta+2.717\beta^{2} \tag{8.3-72}$$

$$\alpha=0.75 \tag{8.3-73}$$

where

β = (T - 500)/100

T = Temperature, °C

The rate, R, and incubation, τ, parameters needed in Eq. (8.3-67) have several options which are controlled through the input parameters IRATE and ITAU. The confidence limits on the use of these equations suggest the use of upper and lower bounds on R and τ. Nominal values are obtained with no multipliers to these parameters. The input parameters IRATE and ITAU can be used to select the specified bound as illustrated below. Input values of zero give the nominal parameters. If cladding swelling does not occur, the values of ITAU and IRATE can be set to bypass the swelling calculation.

Limits on steady-state swelling rate parameter, R:

Upper bound (IRATE = 1)

$$F=1.3+2.5\exp\left[-\left(T-350\right)^{2}\times10^{-4}\right]+1.7\exp\left[-\left(T-650\right)^{2}\times10^{-3}\right] \tag{8.3-74}$$

where

T = Temperature, °C

$$R_{\mathrm{u}}=R\times F \tag{8.3-75}$$

Lower bound (IRATE = -1)

$$R_{1}=R\times0.70 \tag{8.3-76}$$

No swelling (IRATE = -2)

Limits on incubation parameter τ:

Upper bound (ITAU = 1)

$$\tau_{\mathrm{u}} = \tau \times 1.30 \tag{8.3-77}$$

Lower bound (ITAU = -1)

$$\tau_{\mathrm{l}} = \tau \times 0.70 \tag{8.3-78}$$

No swelling (ITAU = -2)

These equations are solved in the subroutine CLADSW.

8.3.6 Fission-gas Plenum Pressure

As the volatile fission products are released from the fuel, they enter the free volume associated with the fuel pin and are assumed to mix homogeneously with the gases already present. The free volumes considered are the fabricated fission gas plenum, the central fuel void not associated with a central molten fuel cavity, the fuel-cladding gap, and the crack volume within the fuel. A homogeneous ideal-gas mixture that is in pressure equilibrium is assumed to form.

The total number of moles of fission gas and helium is known prior to the pressure calculation, although the distribution is not known.

$$n_{T_l} = n_{\mathrm{T}}^{\mathrm{fp}} + n_{\mathrm{T}}^{\mathrm{He}} \tag{8.3-79}$$

where

n_{T} = Total moles of gas

$n_{\mathrm{T}}^{\mathrm{fp}}$ = Moles of fission product gas

$n_{\mathrm{T}}^{\mathrm{He}}$ = Moles of helium

The amount of helium is known from the fill gas pressure, the fraction that is not helium, and the reference temperature geometry. The amount of helium also contains the amount released as porosity migration occurs (Section 8.3.1). The number of moles of fission gas is known from the fission gas release calculation (Section 8.3.3) and the non-helium initial fill gas.

If all the free volume exists at the same pressure, P_{g}, then the number of moles of gas at any specific free volume location can be determined from the ideal gas law.

$$n_1 = \frac{P_{\mathrm{g}}}{R}\left(\frac{V_{\mathrm{i}}}{T_{\mathrm{i}}}\right) \tag{8.3-80}$$

where

n_{i} = Moles of gas in free volume i

P_{g} = Pressure of the gas, Pa

R = Ideal-gas constant, J K^{-1} $(g\text{-}mole)^{-1}$

V_{i} = Volume of free volume i, m^3

T_{i} = Temperature of free volume i, K

This basic relationship can be used to determine the moles of gas associated with the types of free volumes listed above. For the plenum, there exists only one volume and temperature.

$$n_{\mathrm{p}} = \frac{P_{\mathrm{g}}}{R}\left(\frac{V_{\mathrm{p}}}{T_{\mathrm{p}}}\right) \tag{8.3-81}$$

where

n_{p} = Total number of moles in the plenum

V_{p} = Volume of the fission gas plenum, m^3

T_{p} = Temperature of the fission gas plenum, K

The central fuel void may exist over a number of axial segments due to as-fabricated porosity or the fuel initially fabricated with a central hole. Therefore, each axial segment contributes to the number of moles.

$$n_{\mathrm{v}} = \frac{P_{\mathrm{g}}}{R}\left(\sum_{\mathrm{j=1}}^{\mathrm{MZ}} \frac{V_{\mathrm{v,j}}}{T_{\mathrm{v,j}}}\right) \tag{8.3-82}$$

where

n_{v} = Total number of moles in central void

$V_{\mathrm{v,j}}$ = Volume of central void in axial segment j, m^3

$T_{\mathrm{v,j}}$ = Temperature of the central void in axial segment j, K

MZ = Number of axial nodes

If an axial segment has a central void included as part of the molten cavity, its contribution is not included in Eq. (8.3-82).

The fuel-cladding gap can also contain the released gases. It also exists over a number of axial nodes, and its contribution to the total moles of gas is a summation over all axial segments.

$$n_{\mathrm{g}} = \frac{P_{\mathrm{g}}}{R}\left(\sum_{\mathrm{j=1}}^{\mathrm{MZ}} \frac{V_{\mathrm{g,j}}}{T_{\mathrm{g,j}}}\right) \tag{8.3-83}$$

where

n_{g} = Total number of moles in fuel-cladding gap

$V_{\mathrm{g,j}}$ = Volume of fuel-cladding gap in axial segment j, m^3

$T_{\mathrm{g,j}}$ = Temperature of the fuel-cladding gap, $K = 0.5\left(T_{\mathrm{f,j}} + T_{\mathrm{c,j}}\right)$

$T_{\mathrm{f,j}}$ = Fuel surface temperature in axial segment j, K

$T_{\mathrm{c,j}}$ = Inner cladding surface temperature in axial segment j, K

The volume associated with the crack volume requires a summation over both the radial extent of cracking and the axial segments. In a given axial segment, the volumes and temperatures change with the radius.

$$n_{\mathrm{k}} = \frac{P_{\mathrm{g}}}{R}\left[\sum_{\mathrm{j=1}}^{\mathrm{MZ}}\left(\sum_{\mathrm{i=IETA}}^{\mathrm{NT}} \frac{V_{\mathrm{k,i,j}}}{T_{\mathrm{k,i,j}}}\right)\right] \tag{8.3-84}$$

where

n_{k} = Total number of moles in the crack volume

$V_{k,i,j}$ = Volume of cracks in radial cell i of axial segment j, m^3

$T_{k,i,j}$ = Temperature of fuel in radial cell i of axial segment j, K

IETA = Innermost cracked fuel node

NT = Total number of radial fuel nodes

It is assumed that the gas in the cracks of the fuel is in thermal equilibrium with the fuel in the same cell.

Since the total number of moles must reside in the free volumes considered, Eq. (8.3-79) can be combined with Eq. (8.3-81) through Eq. (8.3-84)

$$n_T = \frac{P_g}{R}\left[\frac{V_p}{T_p}\sum_{j=1}^{MZ}\frac{V_{v,j}}{T_{v,j}} + \sum_{j=1}^{MZ}\frac{V_{g,j}}{T_{g,j}} + \sum_{j=1}^{MZ}\left(\sum_{i=IETA}^{NT}\frac{V_{k,i,j}}{T_{k,i,j}}\right)\right] \tag{8.3-85}$$

Everything in Eq. (8.3-85) is known except the gas pressure, therefore the equation can be solved to find the pressure.

$$P_g = \frac{n_T R}{\left[\frac{V_p}{T_p}\sum_{j=1}^{MZ}\left(\frac{V_{v,j}}{T_{v,j}} + \frac{V_{g,j}}{T_{g,j}} + \sum_{i=IETA}^{NT}\frac{V_{k,i,j}}{T_{k,i,j}}\right)\right]} \tag{8.3-86}$$

Once the uniform pressure has been determined from Eq. (8.3-86), and using the known ratio of fission product moles to helium moles with Eq. (8.3-79), the number of moles of fission gas and helium can be found in any of the free volumes.

These calculations are performed in the subroutine PRESPL.

8.3.7 Molten Cavity Pressurization

Prior to fuel melting, the central void and plenum are assumed to be in pressure equilibrium. Once melting has begun, DEFORM-4 provides for three different methods for calculating the molten cavity pressure: (1) the cavity under consideration extends axially only over the range of segments where melting has occurred, (2) the cavity extends over all axial segments, and (3) each axial segment is considered a separate cavity. These options are controlled by the input variable IMELTV.

IMELTV	*Description*
0	Molten cavity extends only over the axial extent of fuel melting
1	Molten cavity extends over all axial segments
2	Each axial segment treated as a separate control volume

If a central void exists before fuel melting, then the first two options give similar results. The second option presents a more mechanistic approach to the molten cavity. As melting begins, it is expected that communication would exist all along the central void. As the available volume for the helium and fission gas associated with one axial segment is decreased because of fuel melting and thermal expansion, the excess gas would move to other segments to produce a balanced pressure all along the central void. This model assumes that the transient time scales are long enough to allow this redistribution process.

The third option is included for the study of extremely fast transients. If the time scale is short enough to preclude material redistribution, then each axial segment is assumed to act as a separate molten cavity. A separate pressure is calculated for each segment, and axial pressure differentials are produced. The effects of these axial differentials on the radial mechanics solution in the cladding can then be studied to determine the sensitivity of cladding failure location.

In addition to these options, the effects of fuel moving into the cracks and pressurizing them to the same level as the molten cavity can be studied through the input parameter IROR. If this parameter is set to 1, the crack volume is included in the molten cavity volume and the cavity pressure acts directly on the cladding, see Section 8.2.3. Because DEFORM-4 does not consider mass transport between axial segments or radial cells, the actual movement of fuel cannot take place. The consideration of the crack volume does allow the macroscopic effect of volume increase in the molten cavity by movement of molten fuel out of the cavity into crack volume to be considered.

When the PINACLE module is initiated, which describes the pre-failure in-pin molten fuel relocation, it provides the molten cavity pressure that is a boundary condition for the DEFORM calculation. This module replaces the CAVITY subroutine in DEFORM. DEFORM and PINACLE work interactively with PINACLE providing the temperatures and molten cavity pressures for DEFORM, and DEFORM determining the fuel pin thermal/mechanical response to changes in these conditions and returning to PINACLE the new axial and radial node locations.

The following calculations are performed in the subroutine CAVITE.

8.3.7.1 Incremental Melt Fraction Ratio

The radial extent of the molten cavity is determined by the relationship between the radial cell temperature and the solidus temperature. Melting of the cell is assumed to begin when the solidus temperature is reached, and be complete when the liquidus temperature is attained.

$$f_{\mathrm{m,i}}\left(t+\Delta t\right)=\frac{\left(T_{2,\mathrm{i}}-T_{\mathrm{s}}\right)}{\left(T_{l}-T_{\mathrm{s}}\right)} \tag{8.3-87}$$

where

$f_{\mathrm{m,i}}$ = Melt fraction of radial cell i

t = Time at beginning of the time step, s

Δt = Time step length, s

$T_{2,\mathrm{i}}$ = Temperature of radial cell i at end of time step, K

T_{S} = Solidus temperature, K

T_{l} = Liquidus temperature, K

As the cell melts, the amount of material considered to be in the molten region of the cell is directly related to the melt fraction, $f_{\mathrm{m,i}}$. However, the original fission gas content, etc., is not saved from one time step to the next, but the values at the end of the time step are determined, so a simple relationship between melt fraction change and material distribution between the molten state vs the solid state based on the current melt fraction cannot be used. It is necessary to develop an approach based on the solid fuel quantities at the beginning of the time step and the melt fraction change during the time step. This relationship is known as the incremental melt fraction ratio. With this approach, the cell can be considered to melt into the molten cavity in an incremental fashion rather than adding the node to the cavity all at one time when some arbitrary melt fraction has been reached.

Prior to melting, a radial cell contains some predetermined amount of a specific constituent that is important to the cavity pressurization, such as fission gas, as-fabricated gas and volume, crack gas and volume, etc. At the first step where melting occurs, some part of this is transferred from the solid region into the molten region.

$$F_{\mathrm{m,i}}\left(1\right)=F_{\mathrm{m,i}}\times F_{\mathrm{o,i}} \tag{8.3-88}$$

$$F_{\mathrm{s,i}}\left(1\right)=F_{\mathrm{o,i}}\times\left[1-f_{\mathrm{m,i}}\left(1\right)\right] \tag{8.3-89}$$

where

$F_{\text{m,i}}(1)$ = Amount of constituent in the molten region at end of time step 1 in radial cell i

$F_{\text{o,i}}$ = Total amount of constituent in solid fuel prior to melting

$F_{\text{s,i}}(1)$ = Amount of constituent remaining in solid region at end of time step 1 in radial cell i

At end of the second time step with melting, the amounts in the molten and solid regions are related to the new time step melt fraction

$$F_{\text{m,i}}(2) = F_{\text{o,i}} \times f_{\text{m,i}}(2) \tag{8.3-90}$$

$$F_{\text{s,i}}(2) = F_{\text{o,i}} \times [1 - f_{\text{m,i}}(2)] \tag{8.3-91}$$

Eq. (8.3-90) and Eq. (8.3-91) can be rewritten in terms of the known quantities $F_{\text{m,i}}(1)$ and $F_{\text{s,i}}(1)$ rather than the now unknown, $F_{\text{o,i}}$ by rearrangement of Eq. (8.3-88) and Eq. (8.3-89).

$$F_{\text{m,i}}(2) = F_{\text{s,i}}(1) \frac{f_{\text{m,i}}(2)}{[1 - f_{\text{m,i}}(1)]} \tag{8.3-92}$$

$$F_{\text{s,i}}(2) = F_{\text{s,i}}(1) \frac{[1 - f_{\text{m,i}}(2)]}{[1 - f_{\text{m,i}}(1)]} \tag{8.3-93}$$

As the process continues, the relationship between the constituents and melt fractions at a specific time can be generalized.

$$F_{\text{m,i}}(t + \Delta t) = F_{\text{s,i}}(t) \frac{f_{\text{m,i}}(t + \Delta t)}{[1 - f_{\text{m,i}}(t)]} \tag{8.3-94}$$

$$F_{\text{s,i}}(t + \Delta t) = F_{\text{s,i}}(t) \frac{[1 - f_{\text{m,i}}(t + \Delta t)]}{[1 - f_{\text{m,i}}(t)]} \tag{8.3-95}$$

The change in the constituent during the time step can be determined by subtracting the results of two time steps and making use of the generalized form of Eq. (8.3-89).

$$\Delta F_{\text{s,i}} = F_{\text{s,i}}(t + \Delta t) - F_{\text{s,i}}(t) = -F_{\text{s,i}}(t) \frac{[f_{\text{m,i}}(t + \Delta t) - f_{\text{m,i}}(t)]}{[1 - f_{\text{m,i}}(t)]} \tag{8.3-96}$$

$\Delta F_{\text{s,i}}$ = Change in constituent in solid fuel in radial cell i that takes place during the time step

The same procedure applied to the change in the molten region constituent results in the negative of Eq. (8.3-96), as would be expected for conservation.

The considerations leading to Eq. (8.3-96) define what is known as the incremental melt fraction ratio.

$$R_{\text{IMF}} = \frac{[f_{\text{m,i}}(t + \Delta t) - f_{\text{m,i}}(t)]}{[1 - f_{\text{m,i}}(t)]} \tag{8.3-97}$$

where

R_{IMF} = Incremental melt fraction ratio

This ratio is used in the considerations given below.

8.3.7.2 Gas Release on Melting

When melting begins in an axial segment, there are five constituents from which gas and volume may be moved into the molten cavity: (1) the initial central void, (2) the grain boundaries, (3) intragranular, (4) the as-fabricated porosity, and (5) the fuel cracks. Each of these can affect the molten cavity pressurization through the gas and volume associated with their inclusion and will be described below. In the discussion in the next five sections, each constituent is treated as a single adjustment to the cavity gas and volume consideration, while in actuality they are accumulated for the total change to the cavity.

8.3.7.2.1 Central Void Considerations

Prior to melting, the central void is considered to be in equilibrium with the rest of the free volume within the pin. The number of moles of helium and fission gas associated with the central void in each axial segment varies depending on the volume, temperature, and pressure, and was discussed in Section 8.3.6. The amount of this gas, therefore, changes during each time step throughout the pretransient and transient up to fuel melting initiation.

Upon initial melting, the gas associated with the defined central molten cavity, see definition of IMELTV above, is fixed at its last value based on equilibrium with the plenum and associated free volume within the pin. This gas and its volume is then considered part of the molten cavity and is removed from the plenum pressure considerations. It is implicitly assumed that melting initiates the production on a molten cavity region, which acts as a bottle that does not communicate with the plenum.

This central void consideration defines the initial state associated with the molten cavity.

8.3.7.2.2 Grain Boundary Gas

As described in Section 8.3.3 and Section 8.3.4 above, the fission gas is assumed to exist in grain boundary bubbles that cause fuel swelling, and intragranular gas within the fuel matrix. Upon fuel melting, the grain boundary gas and its volume is assumed to be released immediately into the molten cavity. Based on the discussion in Section 8.3.7.1, Eq. (8.3-96) and Eq. (8.3-97) can be used to determine the gas released from the grain boundary to the molten cavity.

$$\Delta M_{\text{gb,i,j}}\left(\Delta t\right) = -M_{\text{gb,i,j}}\left(t\right) R_{\text{IMF,i,j}} \tag{8.3-98}$$

$$\Delta M_{\text{gb,i,j}}\left(\Delta t\right) = -M_{\text{gb,i,j}}\left(t\right) R_{\text{IMF,i,j}} \tag{8.3-99}$$

where

$\Delta M_{\text{gb,i,j}}\left(\Delta t\right)$ = Change in moles of fission gas on the grain boundary during the time step in radial cell i of axial segment j

$M_{\text{gb,i,j}}\left(t\right)$ = Moles of fission gas on the grain boundary at beginning of the time step for radial cell i of axial segment j

$\Delta M_{\text{c,gb}}\left(\Delta t\right)$ = Moles of fission gas added to the molten cavity from the grain boundaries during the time step

$R_{\text{IMF,i,j}}$ = Incremental melt fraction ratio for radial cell i of axial node j

The volumes associated with the grain boundary gas is assumed to be moved to the molten cavity in the same proportion as the moles.

$$\Delta V_{\text{gb,i,ij}}\left(\Delta t\right) = -V_{\text{gb,i,j}}\left(t\right) R_{\text{IMF,i,j}} \tag{8.3-100}$$

$$\Delta V_{\text{c,gb}}\left(\Delta t\right) = -\Delta V_{\text{gb,i,j}}\left(\Delta t\right) \tag{8.3-101}$$

where

$\Delta V_{\text{gb,i,j}}\left(\Delta t\right)$ = Change in volume of the fission gas on the grain boundaries during the time step in radial cell i of axial segment j, m^3

$V_{\text{gb,i,j}}\left(t\right)$ = Volume of gas on grain boundaries at beginning of the time step for radial cell i of axial segment j, m^3

$\Delta V_{\text{c,gb}}\left(\Delta t\right)$ = Volume added to molten cavity from grain boundaries during the time step, m^3

As a cell progresses from initial melting to fully molten, all the grain boundary gas and volume is moved into the molten cavity.

Eq. (8.3-98) through Eq. (8.3-101) all refer to a single radial node. The total amount of gas and volume moved into association with the molten cavity is found through a summation of these equations over all melting radial cells at all axial segments in the cavity. Combining Eq. (8.3-98) and Eq. (8.3-99), and Eq. (8.3-100) and Eq. (8.3-101),

$$\Delta M_{\text{c,gb}}^{T} = \sum_{\text{j=jcavb}}^{\text{jcavt}} \left[\sum_{\text{i=1}}^{iz-1} \left(M_{\text{gb,i,j}} \left(t \right) \times R_{\text{IMF,i,j}} \right) \right] \tag{8.3-102}$$

$$\Delta V_{\text{c,gb}}^{T} = \sum_{\text{j=jcavb}}^{\text{jcavt}} \left[\sum_{\text{i=1}}^{iz-1} \left(M_{\text{gb,i,j}} \left(t \right) \times R_{\text{IMF,i,j}} \right) \right] \tag{8.3-103}$$

where

$\Delta M_{\text{c,gb}}^{T}$ = Total moles of gas added to molten cavity from the grain boundaries during the time step

$\Delta V_{\text{c,gb}}^{T}$ = Total volume of grain boundary gas added to molten cavity during the time step, m^3

j_{cavb} = Axial segment number at bottom of molten cavity

j_{cavt} = Axial segment number at top of molten cavity

iz = Radial boundary number that defines the boundary between the molten and solid region

8.3.7.2.3 Intra-Granular Gas

While the grain boundary gas is assumed to move instantaneously into the molten cavity, the intragranular gas is assumed to be tied-up in very small bubbles or interstitially located, so a time is required for the coalescence and buoyancy forces to release this gas into the cavity. Therefore, there can be gas in the molten fuel region which has not been released into the molten cavity.

The intra-granular gas is transferred from the solid fuel into the molten fuel based on the considerations presented in Section 8.3.7.1.

$$\Delta M_{\text{m,ig}} \left(\Delta t \right) = -M_{\text{ig,i,j}} \left(t \right) R_{\text{IMF,i,j}} \tag{8.3-104}$$

$$M_{\text{ig,i,j}} \left(t + \Delta t \right) = M_{\text{ig,i,j}} \left(t \right) - \Delta M_{\text{m,ig}} \left(\Delta t \right) \tag{8.3-105}$$

$$M_{\text{m,ig}} \left(t + \Delta t \right) = M_{\text{m,ig}} \left(t \right) + \Delta M_{\text{m,ig}} \left(\Delta t \right) \tag{8.3-106}$$

where

$\Delta M_{\text{m,ig}} \left(\Delta t \right)$ = Moles of intra-granular fission gas transferred from the solid to the molten fuel during the time step for the radial cell

$M_{\text{ig,i,j}} \left(t \right)$ = Moles of intra-granular fission gas at beginning of the time step in radial cell i of axial segment j

$M_{\text{ig,i,j}} \left(t + \Delta t \right)$ = Moles of intra-granular fission gas at end of time step

$M_{\text{m,ig}} \left(t + \Delta t \right)$ = Moles of gas retained in molten fuel at end of time step

The intra-granular gas is assumed to have no volume associated with it, so no volume movement occurs.

The release of the gas retained within the molten fuel to the molten cavity through coalescence and buoyancy effects is assumed to occur at a rate proportional to the amount of gas present.

$$\frac{\text{dM}_{\text{m,ig}}}{\text{dt}} = -\tau_{\text{g}} M_{\text{m,ig}} \tag{8.3-107}$$

where

τ_g = Time constant for coalescence release, s^{-1}

Eq. (8.3-107) is integrated over the time step to determine the release from the molten fuel to the molten cavity.

$$\Delta M_{c,ig}(\Delta t) = M_{m,ig}(t+\Delta t)\left[1-\exp\left(-\Delta t \times \tau_g\right)\right] \tag{8.3-108}$$

$$\Delta M'_{m,ig}(t+\Delta t) = M_{m,ig}(t+\Delta t) - \Delta M_{c,ig}(\Delta t) \tag{8.3-109}$$

where

$\Delta M_{c,ig}(\Delta t)$ = Moles of intra-granular gas added to molten cavity during the time step

$M'_{m,ig}(t+\Delta t)$ = Moles of intra-granular gas remaining in the molten fuel at the end of the time step after both transfer from solid fuel and release to the cavity

Because this varies with each radial cell and axial segment, the total change is found through the double summation as with Eq. (8.3-102) above.

$$\Delta M^T_{c,ig} = \sum_{j=jcavb}^{jcavt}\left[\sum_{i=1}^{iz-1}\left(\Delta M_{c,ig}\right)\right] \tag{8.3-110}$$

where

$\Delta M^T_{c,ig}$ = Total moles of gas added to molten cavity from the intragranular gas during the time step

8.3.7.2.4 As-Fabricated Gas

Like the grain boundary gas, the gas and volume associated with any residual as-fabricated porosity, see Section 8.3.1, is assumed to be released to the molten cavity instantaneously upon melting.

$$\Delta M_{c,af}(\Delta t) = M_{af,i,j}(t)\, R_{IMF,i,j} \tag{8.3-111}$$

$$M_{af,i,j}(t+\Delta t) = M_{af,i,j}(t) - \Delta M_{c,af}(\Delta t) \tag{8.3-112}$$

$$\Delta V_{c,af}(\Delta t) - V_{af,i,j}(t)\, R_{IMF,i,j} \tag{8.3-113}$$

$$V_{af,i,j}(t+\Delta t) = V_{af,i,j}(t) - \Delta V_{c,af}(\Delta t) \tag{8.3-114}$$

where

$\Delta M_{c,af}$ = Moles of as-fabricated, retained helium added to the molten cavity

$M_{af,i,j}$ = Moles of as-fabricated, retained helium that remains in the solid fuel of radial cell i of axial segment j

$\Delta V_{c,af}$ = Volume of as-fabricated, retained porosity added to the molten cavity, m^3

$V_{af,i,j}$ = Volume of as-fabricated, retained porosity that remains in the solid fuel of radial cell i of axial segment j

The contributions must be summed over all radial cells of all axial segments considered to be a part of the molten cavity.

$$\Delta M^T_{c,af} = \sum_{j=jcavb}^{jcavt}\left[\sum_{i=1}^{iz-1}\left(M_{af,i,j}(t)\, R_{IMF,i,j}\right)\right] \tag{8.3-115}$$

$$\Delta V^T_{c,af} = \sum_{j=jcavb}^{jcavt}\left[\sum_{i=1}^{iz-1}\left(V_{af,i,j}\ (t)\, R_{IMF,i,j}\right)\right] \tag{8.3-116}$$

where

$\Delta M^{T}_{\text{c,af}}$ = Total moles of gas added to molten cavity from the as-fabricated porosity during the time step

$\Delta V^{T}_{\text{c,af}}$ = Total volume added to the molten cavity from the as-fabricated porosity during the time step

8.3.7.2.5 Fuel Crack Gas

As the radial cell melts into the molten cavity, the gas and volume associated with cracks within the fuel are assumed to be added to the cavity instantaneously. The amount of gas within the cracks depends on the plenum pressure and the temperature of the cell, for all are assumed to be in equilibrium with the plenum.

$$M_{\text{ck,i,j}}(t) = \frac{P_{\text{g}} \times V_{\text{ck,i,j}}(t)}{R \times T_{\text{i,j}}} \tag{8.3-117}$$

where

$M_{\text{ck,i,j}}$ = Moles of gas (fission gas + helium) in the crack volume of radial cell i of axial segment j

P_{g} = Plenum pressure, Pa

$V_{\text{ck,i,j}}$ = Volume of cracks of radial cell i of axial segment j, m^3

$\text{T}_{\text{i,j}}$ = Temperature of radial cell i of axial segment j, m^3

The amount of gas and volume moving into the molten cavity is again determined from the incremental melt fraction ratio and the initial conditions.

$$\Delta M_{\text{c,ck}}(\Delta t) = M_{\text{ck,i,j}}(t)\, R_{\text{IMF,i,j}} \tag{8.3-118}$$

$$M_{\text{ck,i,j}}(t + \Delta t) = M_{\text{ck,i,j}}(t) - \Delta M_{\text{c,ck}}(\Delta t) \tag{8.3-119}$$

$$\Delta V_{\text{c,ck}}(\Delta t) = V_{\text{ck,i,j}}(t)\, R_{\text{IMF,i,j}} \tag{8.3-120}$$

$$V_{\text{ck,i,j}}(t + \Delta t) = V_{\text{ck,i,j}}(t) - \Delta V_{\text{c,ck}}(\Delta t) \tag{8.3-121}$$

where

$\Delta M_{\text{c,ck}}$ = Moles of crack volume gas added to the molten cavity

$\Delta V_{\text{c,ck}}$ = Volume of cracks added to the molten cavity, m^3

The total contributions are then summed over all melting radial cells of all axial segments in the molten cavity.

$$\Delta M^{T}_{\text{c,ck}} = \sum_{\text{j=jcavb}}^{\text{jcavt}} \left[\sum_{\text{i=1}}^{iz-1} \left(M_{\text{ck,i,j}}(t)\, R_{\text{IMF,i,j}} \right) \right] \tag{8.3-122}$$

$$\Delta V^{T}_{\text{c,ck}} = \sum_{\text{j=jcavb}}^{\text{jcavt}} \left[\sum_{\text{i=1}}^{iz-1} \left(V_{\text{ck,i,j}}(t)\, R_{\text{IMF,i,j}} \right) \right] \tag{8.3-123}$$

where

$\Delta M^{T}_{\text{c,ck}}$ = Total moles of crack volume gas added to the molten cavity during the time step

$\Delta V^{T}_{\text{c,ck}}$ = Total volume added to the molten cavity during the time step form cracks, m^3

As crack volume gas is added to the molten cavity, it is removed from the plenum pressure calculation along with its associated volume, thereby maintaining conservation of gas.

8.3.7.3 Fuel Volume Changes

The discussion above covered the movement of gas and volume from various sources into the molten cavity. There are two other effects which will change the volume associated with the molten cavity: (1) fuel volume change on melting, and (2) molten fuel thermal expansion. Any changes due to swelling or contraction of the remaining grain boundary fission gas bubbles are treated in the fuel swelling routine, see Section 8.3.4.

DEFORM-4 contains the function routine RHOF, which determines the theoretical density of the fuel at any given temperature, see Section 8.7.1. Since it is assumed that there is no mass transfer between radial cells or axial segments, the volume change for each cell can be determined

$$\Delta V_{\text{th,i,j}}\left(\Delta t\right) = \left[\frac{1}{\rho\left(T_{\text{2,i,j}}\right)} - \frac{1}{\rho\left(T_{\text{1,i,j}}\right)}\right] F_{\text{i,j}} \tag{8.3-124}$$

where

$\Delta V_{\text{th,i,j}}$ = Volume change caused by temperature changes during the time step for radial cell i of axial segment j, m^3

ρ = Temperature dependent fuel theoretical density, kg/m^3

$T_{\text{2,i,j}}$ = Final temperature of radial cell i of axial segment j, K

$T_{\text{1,j,i}}$ = Initial temperature of radial cell i of axial segment j, K

$F_{\text{i,j}}$ = Mass of fuel in radial cell i of axial segment j, kg

Because the density function already incorporates the changes due to melting over the melting range, there is no need to use the incremental melt fraction ratio.

Summing over all radial cells and axial segments results in the total volume change due to thermal affects in the fuel

$$\Delta V^{T}_{\text{c,th}} = \sum_{\text{j=jcavb}}^{\text{jcavt}} \left[\sum_{\text{i=1}}^{\text{iz}-1} \left(\Delta V_{\text{th,i,j}}\left(\Delta t\right)\right)\right] \tag{8.3-125}$$

where

$\Delta V^{T}_{\text{c,th}}$ = Total volume change in molten cavity due to changes in temperature in the fuel, m^3

8.3.7.4 Molten Cavity Pressure

The molten cavity pressure is determined from the ideal gas law.

$$P_{\text{cav}} = \frac{M_{\text{cav}} R T_{\text{cav}}}{V_{\text{cav}}} \tag{8.3-126}$$

where

P_{cav} = Molten cavity pressure, Pa

M_{cav} = Total moles of gas in molten cavity

T_{cav} = Molten cavity temperature, K

V_{cav} = Molten cavity volume, m^3

The number of moles of gas in the cavity is determined by adding the changes discussed above to the amount present at the start of the time step. The volume is similarly determined.

$$M_{\text{cav}} = M_{\text{c,i}} + \Delta M^{T}_{c,\text{gb}} + \Delta M^{T}_{\text{c,ig}} + \Delta M^{T}_{\text{c,af}} + \Delta M^{T}_{\text{c,ck}} \tag{8.3-127}$$

$$V_{\text{cav}} = V_{\text{c,i}} + \Delta V^T_{\text{c,gb}} + \Delta V^T_{\text{c,af}} + \Delta V^T_{\text{c,ck}} + \Delta V^T_{\text{c,th}} \tag{8.3-128}$$

where

$M_{\text{c,i}}$ = Initial moles of gas present in the molten cavity

$V_{\text{c,i}}$ = Initial volume present in the molten cavity.

It is assumed that the boundary between the outer most cell that is melting and the solid fuel remains stationary during the time step. While this is not necessarily true, appropriate choice of the transient time step length will provide a close link between cavity pressurization and mechanical response to this force without the need to iterate between the cavity pressure and mechanical response parts of the code.

Because of the volume changes occurring as a transient progresses, it is possible for the central void to completely close. The case may even exist where there is more volume needed than exists inside the solid boundary. Because DEFORM-4 does not provide for relocation of fuel between axial segments, the additional material cannot be moved. However, to at least study the pressurization effects, when this situation arises, a volume deficit is calculated.

$$V_{\text{d,j}} = V_{\text{f,j}} - V_{\text{sb,j}} \tag{8.3-129}$$

where

$V_{\text{d,j}}$ = Volume deficit for axial segment j, m^3

$V_{\text{f,j}}$ = Volume the fuel would require at axial segment j, m^3

$V_{\text{sb,j}}$ = Volume available inside the solid fuel boundary, m^3

These values are summed over all axial segments to provide the total volume mismatch.

$$V^T_{\text{d}} = \sum_{\text{j=jcavb}}^{\text{jcavt}} V_{\text{s,i}} \tag{8.3-130}$$

This value is then used to decrease the apparent volume given by Eq. (8.3-128).

$$V^T_{\text{cav}} = V_{\text{cav}} - V^T_{\text{d}} \tag{8.3-131}$$

The temperature used in Eq. (8.3-126) is an averaged value for the cavity. First the radially mass averaged temperature of the molten fuel for each axial segment is determined. These temperatures are then weighted based on the volume available at each axial segment for gas to occupy.

$$T_{\text{ma,j}} = \frac{\sum_{\text{i=1}}^{\text{iz}-1} (T_{\text{2,i,j}} \times F_{\text{i,j}})}{\sum_{\text{i=1}}^{\text{iz}-1} (F_{\text{i,j}})} \tag{8.3-132}$$

$$T_{\text{cav}} = \frac{\sum_{\text{j=jcavb}}^{\text{jcavt}} (T_{\text{ma,j}} \times V_{\text{c,j}})}{\sum_{\text{j=jcavb}}^{\text{jcavt}} (V_{\text{c,j}})} \tag{8.3-133}$$

where

$T_{\text{ma,j}}$ = Radially mass averaged fuel temperature over molten region, K

$T_{\text{2,i,j}}$ = Temperature of radial cell i of axial segment j, K

$F_{\text{i,j}}$ = Mass of radial cell i of axial segment j, kg

$V_{\text{c,j}}$ = Volume of the central void at axial segment j, m^3

Using the values determined from Eq. (8.3-133), Eq. (8.3-131), and Eq. (8.3-127), the molten cavity pressure is calculated from Eq. (8.3-126). This procedure is carried out at the beginning of each time step, so the dimensional changes from the previous time step are used to calculate the molten cavity pressure to be used as the boundary condition for the current time step.

8.3.7.5 Fuel Vapor Pressure

In addition to the pressure from the gases in the molten cavity, the fuel vapor pressure is included. The temperature used is the maximum radially mass-averaged temperature over the axial extent of the cavity. In the case where each axial segment is a separate cavity, the mass-averaged temperature of the segment is used. The fuel vapor pressure terms are added to the gas induced pressure to obtain the total molten cavity pressure.

8.3.8 Fuel-cladding Gap Conductance

The thermal coupling between the cladding and the fuel is important because it affects the fuel and coolant temperatures and thereby, the swelling and thermal expansion of the fuel and cladding. The gap conductance provides this coupling. If the values are high, the fuel temperatures are lower, reducing the fuel swelling, thermal expansion, restructuring, fission-gas release, and stored energy. If the gap conductance is poor due to a large gap or a low conductivity gas mixture in the gap, then all fuel temperatures are raised, enhancing the phenomena mentioned previously. Because of the sensitivity of phenomena to temperature, and therefore gap conductance, it is best to describe the elements that contribute to the heat transfer between the fuel and cladding as mechanistically as possible.

SAS4A contains three options for calculating the gap conductances. In order of decreasing mechanistic consideration they are (1) a modified Ross-Stoute model [8-18], (2) a SAS3D parametric model, and (3) a simple SAS3D inverse-gap-size model. These are discussed below. It is recommended that the modified Ross-Stoute be used because of the more mechanistic nature of its formulation.

The gap conductance is determined at the end of each DEFORM-4 time step, and is used for the fuel-cladding thermal coupling when determining the temperatures at the end of the next time step. The modified Ross-Stoute model is coded in the function HGAP. The other two models are coded in the function HBFND. It is not possible to switch between models during the calculation because of modeling inconsistancies and the large changes that might result.

8.3.8.1 Modified Ross-Stoute Gap Conductance Model

The recommended gap conductance model is a modified Ross-Stoute model [8-18]. This model has been used extensively in such codes as LIFE-III [8-5] and GAPCON [8-19] and is considered the most mechanistic model currently available. The heat transfer through the fuel-cladding interface is considered to consist of three components: (1) conduction through the gas between the fuel and the cladding, (2) radiative transfer between the fuel and cladding surfaces, and (3) solid-to-solid heat transfer if the fuel and cladding are in contact. Even if contact does occur, the model still assumes the existence of a gas gap between the surfaces, a result of the effects of the surface roughnesses, which are input parameters. If there is no contact, then the solid-to-solid component is set to zero. Figure 8.3.2 illustrates the fuel-cladding geometry and considerations for the open and closed gap cases.

8.3.8.1.1 Radiative Heat Transfer

The radiative heat-transfer coefficient, h_{r}, is determined from

$$h_{\mathrm{r}} = \frac{q_{\mathrm{r}}/A_{\mathrm{f}}}{(T_{\mathrm{f}} - T_{\mathrm{c}})} \tag{8.3-134}$$

where

h_{r} = Radiative heat-transfer coefficient, W m^{-2} K^{-1}

q_r = Heat transferred from the hotter to the colder surface, W

A_f = Surface area of the fuel from which q_r is transferred, m^2

T_f = Temperature of the fuel outer surface, the hotter one, K

T_c = Temperature of the cladding inner surface, the colder one, K

For radiative heat transfer between two surfaces,

$$\frac{q_r}{A_f} = \sigma \left[\frac{1}{\varepsilon_f} + \frac{A_f}{A_c} \left(\frac{1}{\varepsilon_c} - 1 \right) \right]^{-1} \left[T_f^4 - T_c^4 \right] \tag{8.3-135}$$

where

σ = Stefan-Boltzmann constant, W m^{-2} K^{-4}

ε_f = Emissivity of the fuel surface

ε_c = Emissivity of the cladding surface

A_c = Inner surface area of the cladding to which the heat is being transferred, m^2

Combining Eq. (8.3-134) and Eq. (8.3-135) provides the following result for the radiative heat-transfer coefficient.

$$h_r = \frac{\sigma \left(T_f^2 + T_c^2 \right) \left(T_f - T_c \right)}{\left[\frac{1}{\varepsilon_f} + \frac{r_f}{r_c} \left(\frac{1}{\varepsilon_c} - 1 \right) \right]} \tag{8.3-136}$$

where

r_f = Outer radius of the fuel, m

r_c = Inner radius of the cladding, m

8.3.8.1.2 Conduction Heat Transfer

The heat-transfer coefficient for conduction through the gas in the fuel-cladding gap is given by

$$h_g = \frac{k_g}{(\Delta r)_g} \tag{8.3-137}$$

where

h_g = Conduction heat-transfer coefficient, W m^{-2} K^{-1}

k_g = Thermal conductivity of the gas in the gap, W m^{-1} K^{-1}

$(\Delta r)_g$ = Total effective fuel-cladding gap size, m

The conductivities of the helium and xenon gases that comprise the mixture in the fuel-cladding interface are given in Section 8.7.11 and Section 8.7.12. The conductivity of this mixture of gas is derived from the kinetic theory of gases [8-20].

$$k_g = \sum_{i=1}^{n} \left[\frac{k_i}{\left(1 + \sum_{\substack{j=1 \\ j \neq i}}^{n} G_{ij} \frac{X_j}{X_i} \right)} \right] \tag{8.3-138}$$

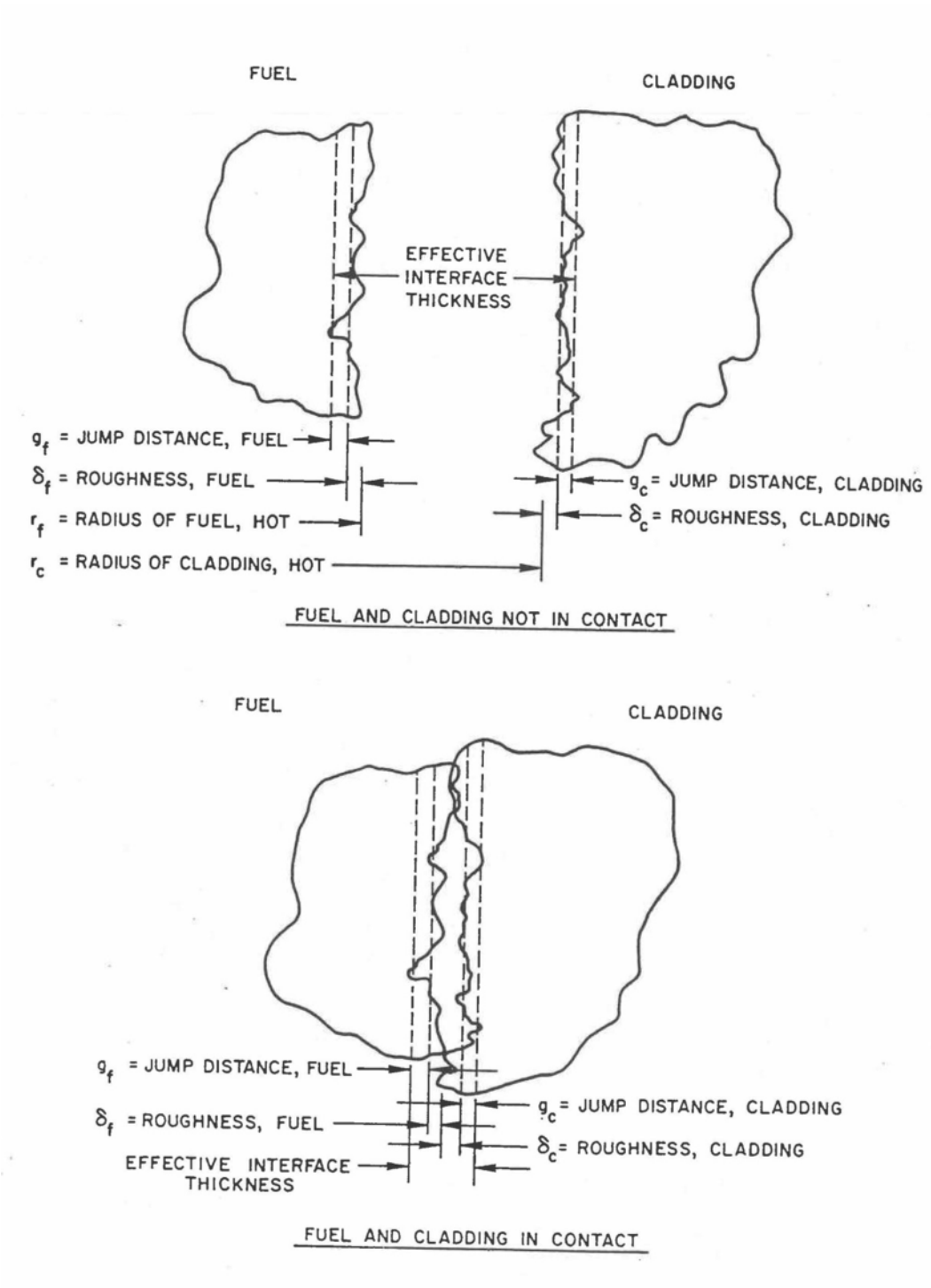


Figure 8.3.2: Geometry of the Fuel-cladding Gap

$$G_{ij} = \frac{1.065}{\sqrt{8}} \left(1 + \frac{M_i}{M_j}\right)^{-1/2} \left[1 + \left(\frac{k_i}{k_j}\right)^{1/2} \left(\frac{M_i}{M_j}\right)^{1/4}\right]^2 \tag{8.3-139}$$

where

k_g = Thermal conductivity of mixture of gases, W m^{-1} K^{-1}

k_i = Thermal conductivity of pure gas component i, W m^{-1} K^{-1}

X_i = Mole fraction of pure gas component i

M_i = Atomic weight of pure gas component i

For the assumed binary mixture of helium and xenon, Eq. (8.3-138) and Eq. (8.3-139) can be made specific with the subscripts h and x referring to the helium and xenon, respectively.

$$G_{hx} = \frac{1.065}{\sqrt{8}} \left(1 + \frac{M_h}{M_x}\right)^{-1/2} \left[1 + \left(\frac{k_h}{k_x}\right)^{1/2} \left(\frac{M_h}{M_x}\right)^{1/4}\right]^2 \tag{8.3-140}$$

$$G_{xh} = \frac{1.065}{\sqrt{8}} \left(1 + \frac{M_x}{M_h}\right)^{-1/2} \left[1 + \left(\frac{k_x}{k_h}\right)^{1/2} \left(\frac{M_x}{M_h}\right)^{1/4}\right]^2 \tag{8.3-141}$$

$$k_g = \frac{k_n}{\left(1 + G_{hx}\frac{X_x}{X_h}\right)} + \frac{k_x}{\left(1 + G_{xh}\frac{X_h}{X_x}\right)} \tag{8.3-142}$$

The effective gap size in Eq. (8.3-137) contains three elements based on the fuel-cladding gap considerations illustrated in Figure 8.3.1. The first is the nominal fuel-cladding gap size. If the fuel and cladding are in contact this value goes to zero. The second consideration is derived from the incomplete energy exchange between the gas atoms and the solid surface of the fuel or cladding. This is modeled as an addition to the gap size. The third consideration is derived from the effects of surface roughnesses. Since the surfaces are not perfectly smooth, the roughness will produce a residual gap through which conduction takes place, as illustrated in Figure 8.3.1.

$$(\Delta r)_g = (r_c - r_f) + (g_c - g_f) + C\,(\delta_c + \delta_f) \tag{8.3-143}$$

where

g_c = Thermal jump distance of the cladding surface, m

g_f = Thermal jump distance of the fuel surface, m

δ_c = Mean surface roughness of the cladding, m

δ_f = Mean surface roughness of the fuel, m

C = Interface pressure dependent factor

The formulation for the temperature jump distances is derived from the kinetic theory of gases [8-20]. The jump distance is directly proportional to the mean free path of the gas in the gap. The extent to which energy exchange occurs when a gas atom strikes the solid surface is defined as the accommodation coefficient, a. This coefficient depends on the species present in the gas, and in the case of a mixture of monoatomic gases is determined from

$$a_{mix} = \frac{\sum a_i X_i M_i^{-1/2}}{\sum X_i M_i^{-1/2}} \tag{8.3-144}$$

where

a_{mix} = Accommodation coefficient of the gas mixture

a_i = Accommodation coefficient of the i-th species of gas in the mixture

X_i = Mole fraction of the i-th species of gas in the mixture

M_i = Molecular weight of the i-th species of gas in the mixture

With the conductivity of the mixture, k_g, defined in Eq. (8.3-142), the jump distance, g, is determined from

$$g = \left(\frac{2 - a_{mix}}{a_{mix}} \right) \frac{(2\pi R_m T)^{1/2} k_g}{(\gamma + 1) C_v P_g} \tag{8.3-145}$$

where

g = Jump distance of the surface under consideration, m

R_m = Gas constant of the gas mixture, J K^{-1}

γ = Ratio of heat capacity at constant pressure to the heat capacity at constant volume

C_v = Heat capacity at constant volume, J kg^{-1} K^{-1}

P_g = Gas pressure in the gap, Pa

From the above equation applied to the fuel and cladding surfaces, the second term in Eq. (8.3-143) can be determined.

The third term concerns the surface roughnesses of the cladding and fuel. The parameter, C, is assumed to depend exponentially on the interface pressure and is given by

$$C = C_o \exp\left(-1.23365 \times 10^{-8} P_i\right) \tag{8.3-146}$$

where

C_o = Input pre-exponential constant, ~ 1.98

P_i = Fuel-cladding interface pressure, Pa

This formulation was determined for use in the GAPCON code [8-19] from the data from Ross and Stoute [8-18]. The value of C is then multiplied by the sum of the mean surface roughnesses of the cladding and the fuel to generate the component of the effective gap resulting from the mismatch of the surface roughnesses. As the interface pressure increases, this gap component becomes smaller because the increased pressure forces a more intimate contact between the surfaces.

8.3.8.1.3 Solid-to-solid Heat Transfer

In addition to conduction through the gap and radiative heat transfer, heat can be passed directly through the solid-to-solid contact points when the gap is closed. Lassmann and Pazdera [8-21] suggest that the coefficient can be described by the following correlation:

$$h_x = \frac{A k_s}{R^{2m-1}} \left(\frac{P_i}{H} \right)^m \tag{8.3-147}$$

where

A, m = Constants determined through experimental comparisons

k_s = Effective thermal conductivity, W m^{-1} K^{-1}

H = Hardness of the softer of the two materials, Pa

R = Square root of the mean squared roughnesses, m

The effective solid conductivity, k_s, is defined by

$$k_s = \frac{2k_c k_f}{(k_c + k_f)} \tag{8.3-148}$$

where

k_c = Thermal conductivity of the cladding at the inner surface, W m^{-1} K^{-1}

k_f = Thermal conductivity of the fuel at the outer surface, W m^{-1} K^{-1}

The square root of the mean squared surface roughnesses, R, is defined by

$$R = \left(\frac{\delta_f^2 + \delta_c^2}{2} \right)^{1/2} \tag{8.3-149}$$

The softer of the two materials is usually the cladding, however, depending on the transient conditions it could be either. Therefore, correlations for both the cladding and fuel have been incorporated, see Section 8.7.13 and Section 8.7.14. The lower of the two values is used in Eq. (8.3-147).

The values of the constants A and m have been determined by Lassmann and Pazdera [8-21] through comparisons with experimental information to be 0.638 for A and 0.67 for m.

8.3.8.1.4 Total Gap Heat-transfer Coefficient

The total heat-transfer coefficient across the fuel-cladding interface is the sum of the three components discussed above.

$$h_{gap} = h_g + h_f + h_s \tag{8.3-150}$$

If the fuel and cladding are not in contact, then the solid-to-solid term is set to zero.

The gap conductance is a parameter that both affects and is affected by changes in the temperatures, dimensions, and gas composition of the fuel-cladding gap. Because of this, the only way to achieve a fully consistent state between the gap conductance, dimensions, and temperatures is through a series of iterations. First an estimate of the gap conductance would be made. Second, the thermal changes for the time step would be calculated. Third, the dimensional and phenomenological changes over the time step would be determined. Fourth, the new state would be used to calculate a new gap conductance. If the initial estimate and the final value agreed within some predetermined error bound, the gap conductance is said to be consistent with the thermal/mechanical state. If the resultant gap conductance was outside the allowed error band, a new estimate would be determined and the entire calculation redone.

While this iteration method would achieve strict consistency, it can lead to long running times for a computer code. Because SAS4A must perform its calculations over many channels and long time periods, it is not possible to use the computational time necessary for the above scenario. Instead, two approximations are used. First, the effect of the new gap conductance estimate on the fuel surface gap temperature is considered. Within the function routine HGAP, iterations are performed over successive estimates of the conductance until the resulting gap conductance value is consistent with the estimated fuel surface temperature for that conductance value, and the resultant fuel and gas conductivities. No dimensional changes are considered, only conductivity effects.

The second method of adjustment has also been employed to smooth the transition between time steps. The new value of gap conductance passed to the thermal routines of SAS4A is an average of the new calculated value and the previous time step value. Experience has shown that using the new calculated value leads to oscillations in the temperatures and dimensions, while this averaging technique produces smoother transitions. In the pre-transient calculation this procedure usually produces less than a 2% difference between the calculated and used gap conductances. This can become much larger during pretransient power changes, but the values

reconverge quickly after a new constant level is attained. The differences are primarily due to dimensional changes not considered in HGAP. In the transient state, the time steps are so short that the dimensional changes are small, resulting in less than a 1% difference in the calculated and used values of gap conductance. These differences are quite acceptable, especially when weighed against the large savings in computational effort.

A final check is made to see if the new value is between the minimum and maximum values allowed, which are input parameters. If the calculated value is outside the bounds, then the appropriate bound is used as the new value. This is useful if a constant gap conductance is required in order to study parametric effects from other models. In ordinary circumstances, these bounds should be set very wide to allow the model to calculate its own value and not be restricted.

8.3.8.2 SAS3D Parametric Model

In the SAS3D code [8-22], a parametric model was employed to determine the gap conductance. This equation is based on the consideration that the gap conductance has some minimum value plus a term that is related to the fuel-cladding gap size and the gas conductivity.

$$h_{\text{gap}} = c_1 + \frac{c_2}{c_3 + (r_{\text{c}} - r_{\text{f}})} \tag{8.3-151}$$

where

c_1, c_{2}`, c_{3}` = Constants

r_{c} = Inner cladding radius, m

r_{f} = Outer fuel radius, m

The second term on the right-hand side of Eq. (8.3-151) approximates the conduction term in the model discussed above (see Section 8.3.8.1.2). If the fuel and cladding are in contact, there still exists a residual gap through which conduction occurs, i.e., c_3 in Eq. (8.3-101). By dividing the numerator and denominator of this conduction term by the constant c_2, the equation form coded in the function HBFND is developed.

$$h_{\text{gap}} = A + \frac{1}{\left\{B + \frac{[(r_{\text{c}} - r_{\text{f}}) + C]}{H}\right\}} \tag{8.3-152}$$

where

A, B, C, H = Input empirical constants

While this model does resemble parts of the model described in Section 8.3.8.1, it also has disadvantages. It is only dependent on the gap size and there is no ability to model the effects of solid-to-solid contact or temperature and gas mixture effects on gas conductivity. Its inclusion in SAS4A does allow for a comparison with SAS3D results while using the same gap conductance parameters.

The values calculated by this model are also restricted by upper and lower bounds.

8.3.8.3 SAS3D Simple Model

The final model depends only on the fuel-cladding gap size. It is included for parametric comparisons with previous SAS3D results. If there is no fuel-cladding gap, the upper bound on the gap conductance is used. If a gap does exist, the conductance is calculated from

$$h_{\text{gap}} = \frac{H}{(r_{\text{c}} - r_{\text{f}})} \tag{8.3-153}$$

where

H = Input constant, W m^{-1} K^{-1}

r_c = Inner cladding surface radius, m

r_f = Outer fuel surface radius, m

If this value falls below the lower bound, the lower bound is used for the next thermal calculation.

8.3.9 Fuel Axial Expansion Reactivity Model

One of the important reactivity feedbacks in reactor safety analysis is the density change caused by the thermal expansion of the fuel and cladding of the pin. Because the magnitude of this effect can depend on the conditions existing at the fuel-cladding interface, a number of options were made available in DEFORM to study this phenomenon. In the previous release version of SAS4A several options were made available; (1) the fuel could be assumed to expand freely, (2) the fuel could be assumed to be constrained against the cladding requiring a force balance between the fuel and cladding, and (3) a mixture of the two could be assumed depending the actual fuel-cladding conditions.

Results from experiments carried out in RAPSODIE [8-23] with oxide fuel have indicated another possibility. Modifications have been made to the EXPAND subroutine of DEFORM to allow for the inclusion of the assumption that the fuel expansion is controlled by the cladding expansion.

If the DEFORM-4 module is used in a calculation, it will perform the calculation for the transient mass redistribution effects from the axial expansion of the fuel and cladding. It is assumed that the fuel and cladding masses within an axial segment remain constant throughout the transient DEFORM-4 calculation. Although axial expansion is calculated for both the upper and lower axial blankets, as well as the active core region, the fuel reactivity is calculated only over the axial segments in the active core region. It is also assumed that the location of the bottom interface of the lower axial blanket does not change. All elevation changes are related to this fixed location.

8.3.9.1 Free Fuel Expansion Controlled Feedback

If the fuel and cladding are assumed to behave independently of each other because of an open fuel-cladding gap or interface conditions that allow the fuel to slip freely along the cladding, the reactivity feedback from axial expansion of the pin will be dominated by the thermal expansion, and thus the temperatures, of the fuel. In oxide fuels where the thermal conductivity of the fuel is low and the fuel-cladding gap conductance is low, this assumption effectively decouples the fuel response from the coolant or cladding temperatures and directly relates the reactivity feedback to the power level.

The consequence of this assumption for TOP transients is an increased negative feedback as the power level increases. In a loss of flow (LOF) transient where the power level may drop during the initial stages, there would be a positive reactivity addition due to the cooling of the fuel as the power decreased. In later stages when boiling produces larger positive reactivity additions, the expansion reactivity would again become negative, particularly during a transient-over-power (TOP) event.

The metal fuel pins would behave in much the same manner, except on a more reduced scale. The high thermal conductivity of the fuel and sodium in the gap produces a much closer coupling between the fuel and coolant temperatures.

8.3.9.2 Free Cladding Expansion Controlled Feedback

The opposite assumption to that discussed in Section 8.3.9.1 above would be to assume that the expansion of the cladding controlled the axial movement of the fuel. The condition necessary for this to be plausible would be an oxide fuel that contains many transverse cracks and fuel connected to the cladding, or a metal fuel connected to the cladding that contains no strength to resist the cladding changes. Some tests and analyses performed with the RAPSODIE reactor [8-23] seem to indicate this type of behavior for oxide fuel.

The consequence of this assumption is to connect the fuel expansion feedback with the coolant, and thereby cladding, temperatures and decouple the feedback from the power level except through its affect on the coolant temperatures. In a TOP scenario, this reduces the magnitude of the reactivity feedback significantly. In the early stages of a LOF event, the effect is to change the magnitude and sign of the feedback from that discussed in Section 8.3.9.1. The feedback would actually be negative as the feedback reached an asymptotic level in a fully voided channel. Differences in magnitude would again occur between the metal and oxide fuels due to the rate of energy transfer to the coolant and the different thermal expansion properties.

This assumption was not included in original version of the SAS4A code but has been added. Since DEFORM calculates both the cladding and fuel response separately under the free expansion assumption (see Section 8.2.4) this assumption was easily added by allowing the cladding expansion to control the expansion of the axial segments.

8.3.9.3 Constrained Fuel/Cladding Expansion Controlled Feedback

The final option available in DEFORM for the calculation of the axial expansion assumes that the fuel and cladding are locked together. The expansion is then determined from the required force balance at the fuel-cladding interface. This model is discussed in Section 8.2.4.

Unlike either of the two previous options, this one requires that both the fuel and cladding responses to the specified transient be considered. It would be expected that the results from these assumptions would fall somewhere between those of the previous two sections since elements of both are included in the considerations. In practice this has been found to be the case. In situations where the fuel and cladding can be assumed to be in contact, such as high burnups with low swelling cladding or medium burnups with the metal fuels, this option would be expected to yield the most credible results. Even when the oxide fuel can be assumed to be heavily cracked, this option may be most accurate because the bowing or twisting of the pin due to the interaction with wire step may cause the cracked fuel to be wedged within the cladding and therefore act as constrained rather than following the cladding expansion, particularly if the number of transverse cracks is small compared to the length of the fuel segments or the pellets can be assumed to have sintered together.

8.3.10 Metal Fuel Modeling

With the renewed interest in uranium metal fuels for breeder reactor applications, there is a need to provide modeling to handle this type of fuel. An initial effort has been made to include important phenomenological features of the metal fuel. This effort involved the use of available DEFORM modeling and incorporation of necessary material properties. Which this effort is not complete, it does provide a basis for evaluating the consequences of the use of metal fuel.

8.3.10.1 Fission Gas Behavior

The buildup of fission gas within the metal fuel produces considerably more fuel swelling than is seen in oxide fuels. In oxide fuels, the strength of the fuel matrix is such that the fission gas bubbles remain small during the irradiation. There is collection of gas on grain boundaries, which leads to fuel swelling, and boundary bubble connecting which produces a local "tunneling" effect that leads to fission gas release.

Recent studies on the growth mechanisms in the analyses of metal fuel swelling [8-24] have indicated that the transient swelling is dominated by grain boundary bubbles, so a single bubble model in an amorphous medium would not be adequate. These grain boundary bubbles also appear to be important for the development of the interlinked porosity in metal fuels. The early work with metal fuels allowed for only low-burnup irradiations because of the extensive swelling, but once it was determined that this "breakway" swelling appeared to be self-limiting, and if enough space was fabricated into the fuel-cladding gap, the result was very little stress on the cladding and high burnups could be achieved [8-25, 8-26]. In order to calculate the steady state behavior of the metal fuel, it is therefore necessary to model this swelling and associated gas release in a consistent manner.

The study of this phenomenon is still underway and not yet developed to the state where an appropriate model could be developed for inclusion into DEFORM. It was therefore necessary to use a more empirical model based on the observed relationship between fuel swelling and fission gas release.

8.3.10.1.1 Closed and Connected Porosity Description

Initially, all created fission gas is assumed to go into fission gas bubbles within the fuel matrix. These bubbles are then allowed to swell in response to any overpressure produced by the bubble pressure acting against the surface tension and hydrostatic pressure [8-27, 8-28]. The amount of swelling allowed is controlled by a time constant determined from the creep rate of the fuel in response to the bubble overpressure. This produces a system of closed porosity within the fuel matrix. The modeling for this creation of closed porosity already existed in DEFORM and the modifications made were to introduce the correct material properties for the metal fuel.

As the closed porosity fraction increases, a point is reached where release would occur. It is assumed that fission gas release is associated with the formation of connected porosity. The release fraction defines the amount of porosity and fission gas that is transferred from closed to open porosity. The calculation of open porosity changes is tied directly to the fission gas release calculation and is explained in more detail below.

8.3.10.1.2 Fission Gas Release

The calculation for the amount of fission gas released from the closed porosity is determined directly from the swelling of the fuel cell. First the volumetric swelling fraction is determined from the fully dense theoretical volume and the actual cell volume, which includes porosity.

$$S_{\mathrm{f}} = \frac{V_{\mathrm{c}} - V_{\mathrm{th}}}{V_{\mathrm{th}}} \tag{8.3-154}$$

where

S_{f} = Volumetric swelling fraction

V_{c} = Actual volume of the fuel cell, m^3

V_{th} = Theoretical volume of the mass of fuel in the cell, m^3

By fitting the swelling vs gas release data from Ref. 8-25 and 8-26, see Figure 8.3.3 the following equation was generated, defining the gas release fraction R_{f}. A least 22.78% swelling is required before the gas is released.

$$R_{\mathrm{f}} = 0.57535 \exp\left(0.33238 S_{\mathrm{f}}\right) - 9.77836 \exp\left(-12.10443 S_{\mathrm{f}}\right) \tag{8.3-155}$$

where

R_f = Fraction of total fission gas released

The total amount of gas released ins therefore the release fraction from Eq. (8.3-155) time the total generated in the node. The amount released during a computational time step is the difference between the total released and the amount released previously.

$$G_r = R_f G_t - (G_t - G_f) \tag{8.3-156}$$

where

G_r = Gas released during the time step, kg

G_t = Total amount of gas produced in the cell, kg

G_f = Gas retained in the fuel assuming no release during the time step, kg

It is then assumed that the release of this amount of gas moves an equivalent fraction of the closed porosity volume to the open porosity.

$$V_r = \frac{G_r}{G_f} V_f \tag{8.3-157}$$

where

V_r = Volume moved from closed to open porosity, m^3

V_f = Volume of fission gas in closed porosity, m^3

These calculations are carried out in the subroutine RELGAS.

8.3.10.2 Metal Fuel Behavior During Melting

8.3.10.2.1 Molten Cavity

The melting of a cell begins as the temperature reaches an input determined temperature, between the solidus and liquidus temperatures. The incremental melt fraction approach developed in Section 8.3.7 is then used to move fuel, gas, and volumes between the solid and molten state. As the temperature of the cell rises above the initial melting temperature, the temperature of the cell rises above the initial melting temperature, increasing fractions of the cell become associated with the molten cavity. When the cell temperature reaches the fuel liquidus temperature, the whole cell is associated with the molten cavity. Since each radial cell is treated separately, an annular melt zone would be correctly treated if it existed in a U-Pu-Zr fuel pin.

8.3.10.2.2 Connected Porosity

Upon melting the connected porosity is released to the molten cavity instantaneously. It is assumed that the bubble morphology associated with this interlinked porosity is such that upon melting, there is no time delay between melting and release of the gas and associated volume to the molten cavity free volume. The connected porosity acts in the same manner as the grain boundary gas in the oxide fuel as far a melting is concerned see Section 8.3.7.2.2. Table 8.3.1 illustrates the similarities and difference between the types of gas associated with the metal fuel and those of the oxide fuel.

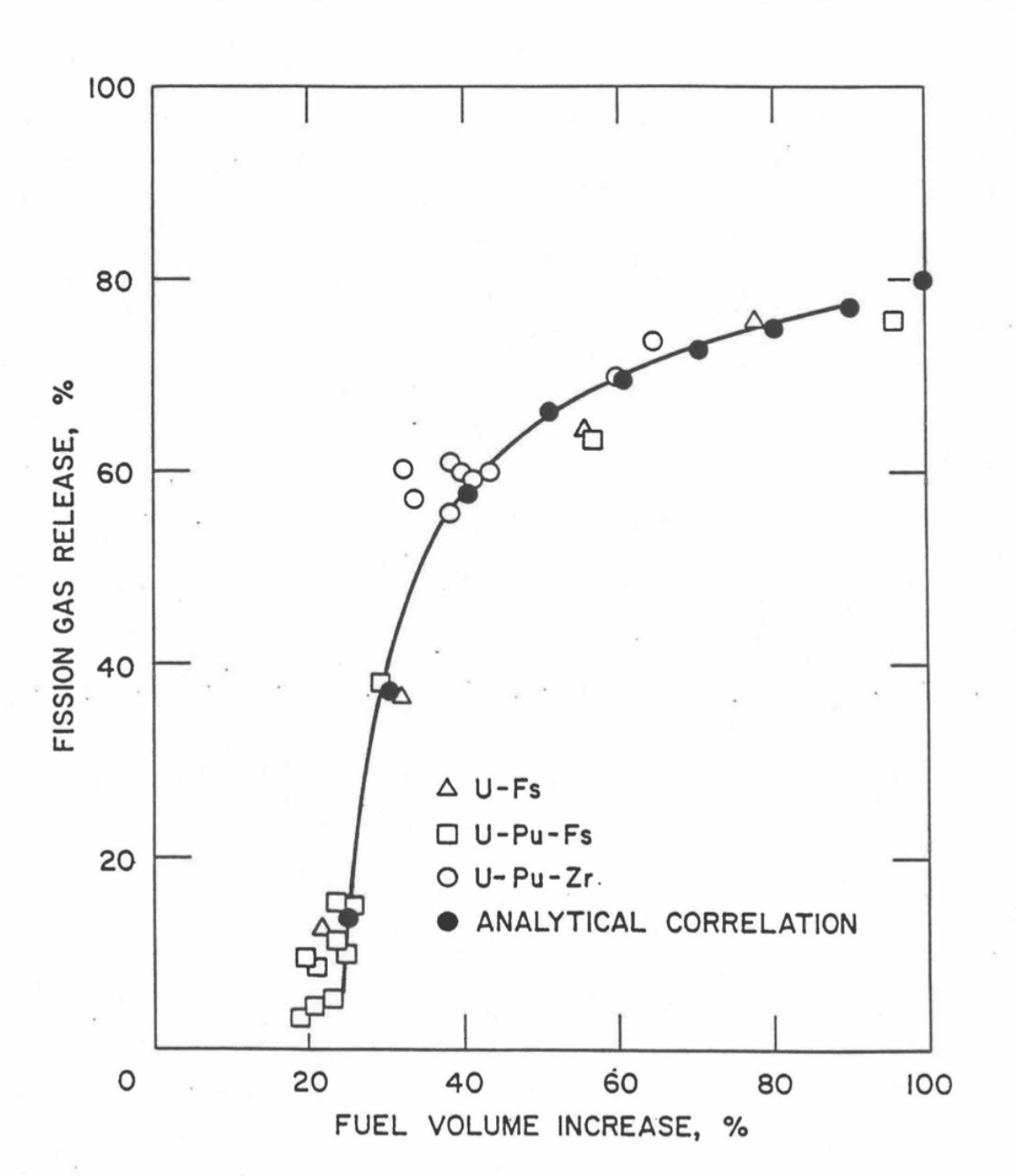


Figure 8.3.3: Effect of Fuel Swelling on Fission Gas Release in Metal Fuels

8.3.10.2.3 Closed Porosity

The closed porosity consisting of fission gas in constrained bubbles is assumed to act like the retained gas in the oxide fuel. Upon melting, these bubbles must move to the free volume in order to coalescence into the molten cavity. This is treated through the use of a time constant for bubble coalescence. Only the retained gas in the molten fuel can undergo this coalescence process (see Section 8.3.7.2.3). Recent results with the model have indicated a need for further refinement by breaking this closed porosity into an intragranular component and a grain boundary component that would be released immediately on melting.

Table 8.3.1: Comparison between Oxide and Metal Fuel Models for Fission Gas Porosity Behavior

Behavior	Oxide Fuel Porosity	Metal Fuel Porosity
Causes fuel swelling	Grain boundary	Closed
Released immediately to molten cavity when fuel melting	Grain boundary	Connected
Release to molten cavity other coalescence delay	Intra-granular	Closed

8.3.10.3 Axial Extrusion in Solid Fuel

One of the reasons for interest in metal fuels is the possibility for axial relocation within the intact cladding. While recent preliminary experimental evidence from the TREAT M2 and M3 tests [8-29] indicate that axial relocation occurs primarily upon fuel melting, the study of solid fuel swelling and extrusion possibilities are valuable for they add to the understanding of the fundamental properties of metal fuel. These phenomena include porosity formation and behavior, fuel creep, fission gas swelling, and fuel-cladding chemical interaction.

8.3.10.3.1 Fission Gas Induced Swelling

DEFORM, for both oxide and metal fuel, calculates the increase or decrease in bubble volume containing trapped fission gas in response to the bubble pressure, the local hydrostatic pressure, the bubble surface tension, and the fuel creep rate. This is described in Section 8.3.4 for the oxide fuel.

In the metal fuel the porosity is divided into open and closed porosity, as described in Section 8.3.10.1.1 above, with the closed porosity assumed to produce the fuel swelling. The same basic approach adopted for the oxide fuel is used for the metal fuel swelling modeling. The amount of retained fission gas and the volume available at the beginning of a time step determine the gas pressure in the closed porosity. This force is counteracted by the local hydrostatic pressure and the bubble surface tension. The mismatch between the bubble pressure and restraining forces is the driving force for expansion (bubble pressure greater than restraining pressure) or contraction (bubble pressure less than restraining pressure) of the closed porosity. If the bubble pressure is the same as the restraining pressure, a state of equilibrium would exist. Therefore, given that the amount of retained gas and the restraining pressure are known, an equilibrium closed porosity volume can be determined that would give a bubble pressure equal to the restraining pressure. This is the volume the closed porosity will tend towards during the time step. However, an instantaneous volume change does not occur because the fuel has a finite creep rate dependent on the mismatch between the initial bubble pressure and restraining pressure, the fuel temperature, and the fission rate. This creep rate defines a time constant associated with the change in bubble volume. The time constant and the current computational time step determine the magnitude of the volume change that takes place as the bubble volume changes from its current volume toward the equilibrium volume.

In the development of the model for the metal fuel, it was found that one of the most sensitive parameters influencing the fuel swelling was the fission gas bubble size used to calculate the surface tension. Because of the low strength of the metal fuel, and the assumption that the interlinked porosity is connected to the fission gas

plenum, the stress state in the fuel is assumed to equal the plenum pressure. Under these conditions the surface tension restraint becomes a critical parameter in the control of fuel swelling. If bubble sizes are small, the surface tension is high and effectively limits the amount of swelling. If bubble sizes are large, then the bubbles expand rapidly to near equilibrium values, introducing rapid swelling of the fuel.

Because there currently exists little information on the development of bubble morphology during the early irradiation of metal fuels, it was necessary to devise a method of providing a bubble size that produced results consistent with the observation that the fuel-cladding gap closes, in metal fuel pins, in the 2-3 atom percent burnup range. This was achieved by providing a bubble radius dependent on the burnup level and then determining a calibration constant to obtain the desired behavior.

$$R_{\mathrm{B}} = C_{\mathrm{R}} B \tag{8.3-158}$$

where

R_{B} = Fission gas bubble radius of closed porosity, m

C_{R} = Calibration constant

B = Burnup, atom percent

While this formulation does not fully adhere to the principle of using mechanistic models employed in DEFORM, it does produce a reasonable initiation to the calculation and provides for development of other models in a consistent fashion.

8.3.10.3.2 Axial Swelling Fraction

Section 8.3.4.1 gives the development of the plane axial swelling strain used in DEFORM. This sam approach is used when considering the metal fuels. However, when the metal fuel comes into contact with the cladding, it is assumed that the fuel strength is not great enough to cause fuel-cladding mechanical interaction. Instead, the volume must be added through axial elongation if the fuel-cladding interface is in such a state that movement can occur, i.e., the temperature of the fuel-cladding interface is above the local eutectic temperature. This movement restriction does not currently exist in DEFORM, but will be operational when DEFORM and SSCOMP are coupled.

After the initial calculation in FSWELL, the outer fuel radius may be greater than the inner cladding radius. The volume that must be accounted for by extra axial expansion is determined by the mismatch between the locations of these two surfaces.

$$\Delta V_{\mathrm{exc}} = \pi A_{\mathrm{H}} \left(r_{\mathrm{ro}}^2 - r_{\mathrm{ci}}^2 \right) \tag{8.3-159}$$

where

ΔV_{exc} = Volume of fuel outside the allowed volume, m^3

A_{H} = Axial segment height including all strains, m

r_{fo} = Outer radius of the fuel including all strains, m

r_{ci} = Inner radius of the cladding including all strains, m

The total volume change calculated to take place radially and that calculated to take place axially, from the initial calculation, are then modified by this volume discrepancy of Eq. (8.3-159)

$$\Delta V_{\mathrm{r,n}} = \Delta V_{\mathrm{ro}} - \Delta V_{\mathrm{exc}} \tag{8.3-160}$$

$$\Delta V_{\mathrm{a,n}} = \Delta V_{\mathrm{a,o}} + \Delta V_{\mathrm{exc}} \tag{8.3-161}$$

where

$\Delta V_{\text{r,n}}$ = New total change in radial volume, m^3

$\Delta V_{\text{r,o}}$ = Old total change in radial volume, m^3

$\Delta V_{\text{a,o}}$ = New total change in axial volume, m^3

$\Delta V_{\text{a,o}}$ = Old total change in axial volume, m^3

These are then used to calculate the new fraction of volume change taking place axially.

$$F_{\text{as}} = \frac{\Delta V_{\text{a,n}}}{\Delta V_{\text{r,n}} + \Delta V_{\text{a,n}}} \tag{8.3-162}$$

where

F_{as} = Fraction of volume change taking place axially

This new factor is then used to determine the radial and axial volume changes for each radial cell, and a new generalized axial swelling strain is calculated following the procedure in Section 8.3.4.1. The new node structure is then determined and the fuel and cladding surfaces are again examined. If a mismatch again occurs, the procedure is repeated until a match is formed within a tolerance of 1.0 x 10^{-12} m. This procedure converges within three to five iterations. Separate thermal/mechanical and swelling strains are maintained for the purpose of comparison. Those calculations are performed in the subroutine FSWELL.

8.3.10.4 Sodium Bond

As the fuel-cladding gap changes, the liquid sodium that fills this gap must move to other locations. A new subroutine, NABOND, has been developed which carries out the accounting of the available volumes and calculates the sodium slug level in the plenum. The temperature of the sodium at a particular elevation is assumed to be the average of the fuel and cladding surface temperatures. The mass of sodium at each axial location is determined from the gap volume and sodium density.

$$M_{\text{s,j}} = \pi A_{\text{H}} \left(f_{\text{ci}}^2 - r_{\text{fo}}^2 \right) \rho_{\text{s,j}} \tag{8.3-163}$$

where

$M_{\text{s,j}}$ = Mass of sodium in fuel-cladding gap at axial segment j, kg

$\rho_{\text{s,j}}$ = Density of liquid sodium at axial segment j, kg/m^3

The total sodium mass in the gap is then determined by summing over all axial segments.

$$M_{\text{st}} = \sum_{\text{j=1}}^{\text{MZ}} M_{\text{s,j}} \tag{8.3-164}$$

where

M_{st} = Total mass of sodium in fuel-cladding gap, kg

MZ = Total number of axial segments in the pin

The sodium mass in the plenum is then found by subtracting the result of Eq. (8.3-164) from the total sodium initially loaded into the pin.

$$M_{\text{sp}} = M_{\text{s1}} - M_{\text{st}} \tag{8.3-165}$$

where

M_{sp} = Mass of bond sodium in the plenum, kg

M_{sl} = Mass of bond sodium initially loaded in the pin, kg

The sodium level in the plenum is then determined.

$$H_{sp} = \frac{M_{sp}}{\rho_{sp} \pi r_p^2} \tag{8.3-166}$$

where

H_{sp} = Height of sodium in the plenum, m

ρ_{sp} = Density of sodium at the plenum temperature, kg/m^3

r_p = Inner cladding radius in the plenum, m

In the calculation of the plenum pressure, this sodium filled volume is removed from the plenum volume available for the released fission gas. The amount released during a computational time step is the difference between the total released and the amount released previously.

8.4 Fuel-pin Failure Criteria

DEFORM-4 contains a number of methods for the determination of fuel-pin failure and failed pin model initiation. These can be divided into two major areas: (1) failure based on specified conditions, and (2) calculated failure correlations. The first method is currently used to produce failure initiation because of the limited amount of integral code validation with the failure correlations. Since these use parameters such as stress and temperature to produce the predictions, it is essential that these be checked against experimental tests to assure the accuracy and appropriateness of their use under various transient conditions. The first method involves parameters used previously with the SAS3D code system, so some familiarity with their use and limitations is assumed. Although these are simple, they can be used in a manner consistent with the SAS3D implementation and available experimental data.

All criteria are cast into the form of a failure fraction, which is the current value for the parameter being used as a failure indicator, i.e., melt fraction, time, etc., divided by the value which would produce failure. If this failure fraction becomes 1, then failure is assumed to occur and the failed pin modeling is initiated. In addition to satisfying the parameter under consideration, the fuel melt fraction must be above a specified level, FMELTM, before PLUTO2 or LEVITATE will be initiated. These calculations are performed in the subroutine CLDFAL.

8.4.1 Input Specified Failure Criteria

Through the use of the input parameters MFAIL, IFAIL, JFAIL, FSPEC, and FMELTM the code can be controlled to initiate failure with the conditions discussed below. MFAIL selects the meaning of FSPEC, IFAIL and JFAIL specify the radial and axial location of the test for failure, and FMELTM is the minimum melt fraction necessary at the location of failure for PLUTO2 or LEVITATE to be initiated. Table 8.4.1 gives the selections currently available in conjunction with DEFORM-4. If IFAIL and JFAIL are not specified, the location of the parameter maximum is selected.

Table 8.4.1: Failure Initiation Options

MFAIL	FSPEC	IFAIL	JFAIL
1	Time	-	Nec [a]
2	Temperature	Nec	Opt [b]
3	Melf fraction	-	Opt
4	Cavity pressure	-	Opt
5	Hoop stress	-	Opt
7	Rip propagation	-	Opt

[a] Nec = Necessary parameter that must be specified.

[b] Opt = Optional parameter. If not specified, the axial level with the maximum value will be chosen.

The use of MFAIL with a value of 7 initiates the failure with a criteria that is consistent with the LEVITATE and PLUTO2 rip propagation considerations, see Section 16.3.1 and Section 14.3.3, respectively. This criterion checks the cladding hoop stress introduced by a molten cavity pressure, for fully cracked fuel, or fuel-cladding mechanical interaction, when solid fuel is present, against the ultimate tensile strength of the cladding to determine if failure has occurred. If the cladding does fail, then the routine checks to see if the molten fuel has reached the crack boundary. Both conditions are necessary for the initiation of the failed fuel routines.

When other failure criteria are used, there may be extensive rip propagation after initiation of the failed fuel model because of the change in criterion.

8.4.2 Calculated Failure Correlations

In DEFORM-4, there exist a number of failure correlations that may be used to determine the time and location of cladding failure. Care should be taken when drawing any conclusions based on these correlations, since the validation effort is in its initial stages with these correlations. In addition, the method of determining cladding stress is not totally consistent between DEFORM-4, a more mechanistic approach, and the correlations, which were based on initial cold dimensions and fill-gas pressures. DEFORM-4 also considers fuel-cladding mechanical interaction rather than just gas pressure.

The use of these failure correlations currently involves an iterative approach, using the failure fractions printed by SAS4A and one of the failure conditions described in Section 8.4.1. First, the case is run with no failure conditions specified and restart files produced at selected intervals. The output is then studied to determine when the failure fraction of interest reaches one, and at which axial segment. One of the conditions discussed above is then selected to initiate failure at the appropriate time and axial segment. A restart case is then run from a restart file produced before the correlation failure condition was reached. In this case, one of the simple failure criteria will be used to initiate failure. The values used in the simple criteria should be chosen to produce failure initiation at the same time and place as was predicted by the correlation criterion during the initial run.

The various correlations are given below. Although the units are inconsistent in many cases, the code has taken this into account and made the necessary adjustments.

8.4.2.1 Biaxial Stress Rupture

This correlation is from the Nuclear Systems Material Handbook [8-17] and is based on unirradiated, 20% cold-worked type 316 SS developmental cladding.

$$\log\theta = -15.22 + 9.5342 \log\left[\log\left(\frac{\sigma^*}{\sigma}\right)\right] \tag{8.4-1}$$

where

θ = Dorn parameter = $t_\mathrm{r} \exp\left(\frac{-Q}{RT}\right)$

t_r = Rupture time, hr

T = Temperature, K

Q = 83,508 cal/mole

R = Universal gas constant

σ = Hoop stress = $\frac{r_\mathrm{o}^2+r_\mathrm{i}^2}{r_\mathrm{o}^2-r_\mathrm{i}^2}P$, ksi

σ^* = 135 ksi

r_o = Clad outer radius

r_i = Clad inner radius

P = Internal gas pressure (or interface pressure), ksi

8.4.2.2 Burst Pressure

This correlation is from the NSMH [8-17] for unirradiated FTR cladding.

$$P = \sum_{\mathrm{i}=0}^{10} a_\mathrm{i}\sigma^i \tag{8.4-2}$$

where

P = Burst pressure, ksi

σ = Hoop stress, as defined above

$a_0 = 1.799988 \times 10^{1}$

$a_1 = 2.866442 \times 10^{-2}$

$a_2 = -3.986012 \times 10^{-4}$

$a_3 = 2.408207 \times 10^{-6}$

$a_4 = -8.090292 \times 10^{-9}$

$a_5 = 1.607218 \times 10^{-11}$

$a_6 = -1.962158 \times 10^{-14}$

$a_7 = 1.487159 \times 10^{-17}$

$a_8 = -6.821934 \times 10^{-21}$

$a_9 = 1.735220 \times 10^{-24}$

$a_{10} = 1.879417 \times 10^{-28}$

8.4.2.3 Transient Burst Temperature

This correlation is from the NSMH [8-17] for unirradiated cladding. It involves two temperature ramp rates, 5.56 and 111.1 K/s. If the rate is between these, a linear interpolation is performed.

$$T_{\mathrm{f}} = 2358.4 - 36.41\sigma + 0.5649\sigma^2 - 3.455 \times 10^{-3}\sigma^3 \tag{8.4-3}$$

for $\dot{T} \leq 5.56$ K/s

$$T_{\mathrm{f}} = 2484.8 - 37.80\sigma + 0.5827\sigma^2 - 3.585 \times 10^{-3}\sigma^3 \tag{8.4-4}$$

for $\dot{T} \geq 111$ K/s

where

T_{f} = Failure temperature, °F

Σ = Hoop stress, as defined above

8.4.2.4 Larson-Miller Life Fraction

This correlation is the LMP life fraction incorporated in the TEMECH computer code [8-30].

$$\log\left(\frac{t_{\mathrm{r}}}{3600}\right) = \frac{\mathrm{LMP}}{1.8T} - C \tag{8.4-5}$$

where

t_{r} = Time to rupture, in seconds

C = Material-dependent constant = 20

LMP = Experimentally determined Larson-Miller parameter

T = Cladding temperature, K

For 20% CW 316 SS cladding, the following correlation is used.

$$\mathrm{LMP}' = 4.6402 - 5.1218 \times 10^{-2}\sigma_{\mathrm{m}} + 7.0417 \times 10^{-4}\sigma_{\mathrm{m}}^2 - 4.1349 \times 10^{-6}\sigma_{\mathrm{m}}^3 \tag{8.4-6}$$

for $0 \leq$ fluence $\leq 1.9 \times 10^{22}$

$$\mathrm{LMP}' = 4.2281 - 2.0469 \times 10^{-2}\sigma_{\mathrm{m}} \tag{8.4-7}$$

for $1.0 \times 10^{22} <$ fluence $\leq 3 \times 10^{22}$

$$\mathrm{LMP}' = 7.488 - 0.138\sigma_{\mathrm{m}} \tag{8.4-8}$$

for 5.56 K/s temperature ramp and fluences between $3.0 - 4.0 \times 10^{22}$

$$\mathrm{LMP}' = 5.285 - 7.778 \times 10^{-2}\sigma_{\mathrm{m}} + 6.027 \times 10^{-4}\sigma_{\mathrm{m}}^2 \tag{8.4-9}$$

for 111.1 K/s temperature ramp and fluences between $3.0 - 4.0 \times 10^{22}$

where

$\mathrm{LMP} = \mathrm{LMP}' \times 10^4$

σ_{m} = Modulus modified hoop stress in ksi = $\sigma_{\mathrm{c}} \frac{E(1033C)}{E(T)}$

These rupture times are then used in the life fraction form as

$$\text{life fraction} = \sum_{\text{i}} \frac{\Delta t_{\text{i}}}{t_{\text{r,i}}} \tag{8.4-10}$$

where

i = Time-step number

Δt_{i} = Time-step length

$t_{\text{r,i}}$ = Time for cladding rupture

8.4.3 Preliminary Metal Fuel Failure Criteria

The consideration of a new fuel type in the context of DEFORM and SAS4A raises the question of the type of failure these pins will experience. Because metal fuel does not produce the same magnitude of fuel-cladding mechanical interaction as the oxide fuel, it is reasonable to assume that the cladding failure will result from a different phenomenon.

Metal uranium forms a low melting point eutectic alloy with both iron and nickel, constituent materials in the cladding alloys used. As the steady state irradiation proceeds, the iron and nickel can be leached out of the cladding, leading to the formation of a low melting point alloy layer adjacent to the cladding surface. Under transient conditions the temperature in this outer layer, at a certain axial location, may become high enough to melt this material. This in turn can produce a failure at this location by accelerating the further thinning of the load bearing cladding. A number of tests with UFs rods were performed and analyzed to develop a correlation for the time to cladding failure [8-31]. This correlation has been incorporated into the FAILUR routine for preliminary use until a more mechanistic model is developed which handles the cladding thinning and subsequent failure from the local pressure.

$$T_{\text{f}} = 9.142 \times 10^4 \left[\frac{T_{\text{a}}}{T_{\text{cut}}}\right]^{-28.495} (1+B)^{-0.54669} \tag{8.4-11}$$

where

T_{f} = Time to failure at the current conditions, s

T_{a} = Temperature at the fuel-cladding interface, °C

T_{eut} = Input eutectic temperature, °C

B = Burnup, atom percent

Once this time has been determined, a life fraction approach is used by dividing the current time step by this failure time and adding it to the previously calculated fractions. When the fraction reaches 1, failure is assured.

$$F_{\text{f,n}} = F_{\text{F,O}} + \frac{\Delta t}{T_{\text{f}}} \tag{8.4-12}$$

where

$F_{\text{f,n}}$ = New life fraction value

$F_{\text{f,o}}$ = Sum of previous life faction values

Δt = Time step, s

This correlation was developed over a narrow range of temperatures so its use should be considered preliminary until a more mechanistic approach is developed.

8.4.4 Failure Modeling Coupling to Fuel Motion Models

In order to correctly assess the accident scenario in a transient that leads to fuel pin failure and subsequent fuel motion, there should be a high degree of consistency between the methods for predicting the initial failure and that used to predict the axial propagation of this failure. This consistency has now been incorporated in the FAILUR subroutine for the option with MFAIL equal to 6. While the actual criterion used is preliminary in nature, the fact that consistency exists between failure and rip propagation is important in order to study how this affects the accident scenario. It is noted that his consistent failure model has been implemented only for an oxide-fuel type failure. A similar consistent failure model for the metal fuel pins will be added in the future.

PLUTO2 and LEVITATE use a rip propagation model that compares the circumferential cladding stress, induced by a pressurized molten cavity acting on cracked fuel, to the ultimate tensile strength of the cladding. A stress greater than the ultimate tensile strength produces failure at the axial location. This same approach has been followed in this failure prediction.

First, the SAS4A/DEFORM cladding node structure is converted to one compatible with to the failed fuel modeling routines. PLUTO2 and LEVITATE divide the cladding into an inner cladding node containing three-fourths of the cladding thickness, and an outer node containing the rest.

$$r_{c1} = r_{ci} + 0.25\,(r_{co} - r_{ci}) \tag{8.4-13}$$

$$r_{c2} = r_{ci} + 0.75\,(r_{co} - r_{ci}) \tag{8.4-14}$$

where

r_{c1} = Outer boundary of old inner cladding cell, m

r_{c2} = Outer boundary of old central cladding cell, m

r_{co} = Outer radius of the cladding, m

r_{ci} = Inner radius of the cladding, m

The new inner cladding cell energy is then determined form weighting factors based on the old cell radii

$$E_{c,a} = \frac{[E_c\,(T_{ci}) \cdot W_1 + E_c\,(T_{cc}) \cdot W_2]}{[W_1 + W_2]} \tag{8.4-15}$$

where

$E_{c,a}$ = Average energy content of the new inner cladding node, J/kg

(E_cT) = Cladding energy as a function of temperature, J/kg

T_{ci} = Inner temperature of the cladding, K

T_{cc} = Central temperature of the cladding, K

$W = r_{c1}^2 - r_{ci}^2$

$W_2 = r_{c2}^2 - r_{c1}^2$

The temperature of the new inner cladding node is then determined.

$$T_{c1} = T_c\,(E_{c,a}) \tag{8.4-16}$$

where

T_{c1} = Temperature of modified inner cladding node, K

$T_c\,(E)$ = Cladding temperature as a function of energy, K

The ultimate tensile strength is then determined on the basis of the new node structure.

$$U_{\text{ts}} = \frac{[3U_{\text{t}}(T_{\text{c1}}) + U_{\text{t}}(T_{\text{co}})]}{4} \tag{8.4-17}$$

where

U_{ts} = Average ultimate tensile strength of the cladding, Pa

$U_{\text{t}}(T)$ = Ultimate tensile strength as a function of temperature, Pa

T_{co} = Cladding outer surface temperature, K

This is the value that determines if failure occurs. If the calculated circumferential stress exclude U_{ts}, failure is assumed to occur.

The calculated stress is determined in two ways, depending on the relationship between the melt boundary and cracked region of the fuel. If a solid annulus of fuel exists between the melted and cracked regions, the calculated fuel-cladding interface pressure is used

$$\sigma_{\text{c}} = \frac{[P_{\text{fci}} r_{\text{ci}} - P_{\text{ext}} r_{\text{co}}]}{[r_{\text{co}} - r_{\text{ci}}]} \tag{8.4-18}$$

where

σ_{c} = Calculated circumferential stress based on a force balance, Pa

P_{fci} = Pressure that exists at the fuel-cladding interface, Pa

P_{ext} = Pressure that exists on the outer cladding surface,

If the melting has proceeded to the cracked boundary, then the cladding stress is determined from a force balance on the cladding assuming an inverse r drop-off in pressure through the solid, cracked fuel.

$$\sigma_{\text{c}} = \frac{[P_{\text{cav}} r_{\text{cav}} - P_{\text{ext}} r_{\text{co}}]}{[r_{\text{co}} - r_{\text{ci}}]} \tag{8.4-19}$$

where

P_{cav} = Central molten cavity pressure, PA

r_{cav} = Radius of the central molten region, m

The failure fraction is then calculated from the cladding stress and ultimate tensile strength.

$$F_{\text{f}} = \frac{\sigma_{\text{c}}}{U_{\text{ts}}} \tag{8.4-20}$$

where

F_{f} = Cladding failure fraction at current time

When this fraction reaches 1, failure of the cladding at the axial segment is assume to occur. However, this does not mean the ejection of fuel into the coolant channel will take place. Besides cladding failure, the molten fuel radius must have reached the cracked fuel radius. Therefore, there are tow conditions to satisfy prior to initiation of the post failure fuel motion modeling; (1) cladding failure, and (2) complete solid fuel cracking.

Once the post failure fuel motion modeling has been initiated, DEFORM is not used. Control is transferred to PLUTO2 or LEVITATE and the same procedure given above is employed by these modules to determine if the cladding failure propagates to other axial segments.

8.4.5 Time-step Control on Approach to Failure

As the conditions necessary for the failure are approached, the maximum time step is reduced to specified levels. This is controlled through the input parameters FIRLIM, SECLIM, and THRLIM, and the associated time steps DTFALl, DTFAL2, and DTFAL3. When the failure fraction reaches FIRLIM, the main time step will be reduced to DTFAL1 or the current time step, whichever is smaller. This same procedure continues through SECLIM and THRLIM. If

$$0 \leq F_{\mathrm{f}} < \mathrm{FIRLIM} \tag{8.4-21}$$

where

F_{f} = Failure fraction for the failure condition being used

then the maximum allowable time step is that defined by the initial transient time-step input parameter DT0.

$$\Delta t = \mathrm{DT0} \tag{8.4-22}$$

where

Δt_{m} = Maximum allowable time-step length, s

If

$$\mathrm{FIRLM} \leq F_{\mathrm{f}} < \mathrm{SECLIM} \tag{8.4-23}$$

then

$$\Delta t_{\mathrm{m}} = \mathrm{DTFAL1} \tag{8.4-24}$$

Similarly, if

$$\mathrm{SECLIM} \leq F_{\mathrm{f}} < \mathrm{THRLIM} \tag{8.4-25}$$

then

$$\Delta t_{\mathrm{m}} = \mathrm{DTFAL2} \tag{8.4-26}$$

or, if

$$\mathrm{THRLM} \leq F_{\mathrm{f}} \leq 1 \tag{8.4-27}$$

then

$$\Delta t_{\mathrm{m}} = \mathrm{DTFAL3} \tag{8.4-28}$$

This procedure serves two functions. First, the reduction in main time step produces a corresponding reduction in the PRIMAR time step, thereby allowing the loop model to continue a stable calculation on the initiation of pin failure.Second, the reduced time step on initiation of failure avoids excessive fuel motion in PLUTO2 and LEVITATE prior to the time these failed pin models can initiate their own time-step control procedures.

8.5 General Method of Solution

DEFORM-4 is coupled to the rest of SAS4A through two main driver subroutines; SSFUEL for the pre-transient calculation, and DFORM3 for the transient-state calculations. In this manner, the DEFORM-4 subroutines

can be considered a separate code module which uses results from the SAS4A thermal/hydraulic calculation, temperatures and coolant pressures, and returns a new set of pin dimensions and properties. A general outline of this interaction scheme is shown in Figure 8.5.1.

In the integration of DEFORM-4 into SAS4A, one of the primary considerations was to minimize the core storage and computational effort necessary when running a multichannel case with a long pre-transient and transient sequence. Based on these considerations, it was decided to avoid the thermalmechanical iteration process and, instead, provide time-step control for up to eight separate pre-transient irradiation periods. This will be discussed in more detail below. In the transient state, the time steps are short enough to avoid the problem of inconsistent results between the thermal and mechanical calculations.

As Figure 8.5.1 shows, SAS4A performs thermal/hydraulic calculations over a specific time step in response to power and coolant conditions. During this time step, the dimensions are assumed to remain constant. SAS4A then passes to DEFORM-4 the temperatures at the beginning and end of the time step, the final coolant pressures, and the length of the time step. DEFORM-4 uses this information to calculate the fuel-pin response to the temperature changes and returns to SAS4A a new set of dimensions consistent with the final thermal/hydraulic conditions. These new dimensions are then used in the calculation of the temperatures at the end of the next time step. In this manner SAS4A and DEFORM-4 march through the entire pre-transient and transient calculation.

The flowchart shown in Figure 8.5.2 illustrates the calling sequence in the pre-transient driver, SSFUEL, and Fig. Figure 8.5.3 shows it in the transient driver, DFORM3. Each phenomenon considered is modularized into a separate subroutine. This makes the inclusion of new models and phenomena a very straightforward process of replacing or modifying specific subroutines. This same procedure has been followed in the incorporation of material properties, as outlined in Section 8.7. A number of the phenomena used in the pre-transient calculation are not included in the transient calculation. Since the time scale of the transient is orders of magnitude less than the pre-transient, phenomena such as pore migration and irradiation-induced cladding swelling are assumed to be inactive, or less than second-order effects. In this manner, the transient calculation is streamlined. If a particular case needs to include these phenomena in the transient, their inclusion is merely a matter of including the appropriate “CALL” statements in the transient driver routine.

For the mechanics calculation, the same routines are used in both the pre-transient and transient calculation. In this way, the pre-transient is considered an extended transient calculation. Since both states use the same routines, the use of computer storage is minimized. Also, since phenomenological changes take place that impact temperatures and dimensions, the pretransient is a special transient. In order to take maximum advantage of this basic structure, the ability to input ability to input up to eight different ramps or steady-state levels is incorporated into the SAS4A pre-transient calculation. It is therefore possible to follow a simplified power history prior to the start of the transient.

In the current version of DEFORM-4, an approach has been adopted which separates the elastic and plastic deformation. These are linked through the fuel-cladding interface pressure. The basic mechanics approach is through a generalized plane strain representation with an elastic cladding. The plastic deformation is derived from two sources. If the fuel-cladding interface pressure produces cladding stresses below the flow stress, plastic deformation can occur through the process of creep. If the stresses induced exceed the flow stress, the cladding is assumed to behave in a perfectly plastic manner and the outer boundary condition on the fuel becomes the pressure that produces cladding stresses equal to the flow stress. In this situation, the radial deformation is determined by the fuel displacement until conditions return to the elastic state.

The calculations are performed on an axial segment by segment basis, starting at the lower axial blanket and continuing upward to the top axial blanket. A complete set of calculations for the response of the axial segment is performed before proceeding to the next segment.

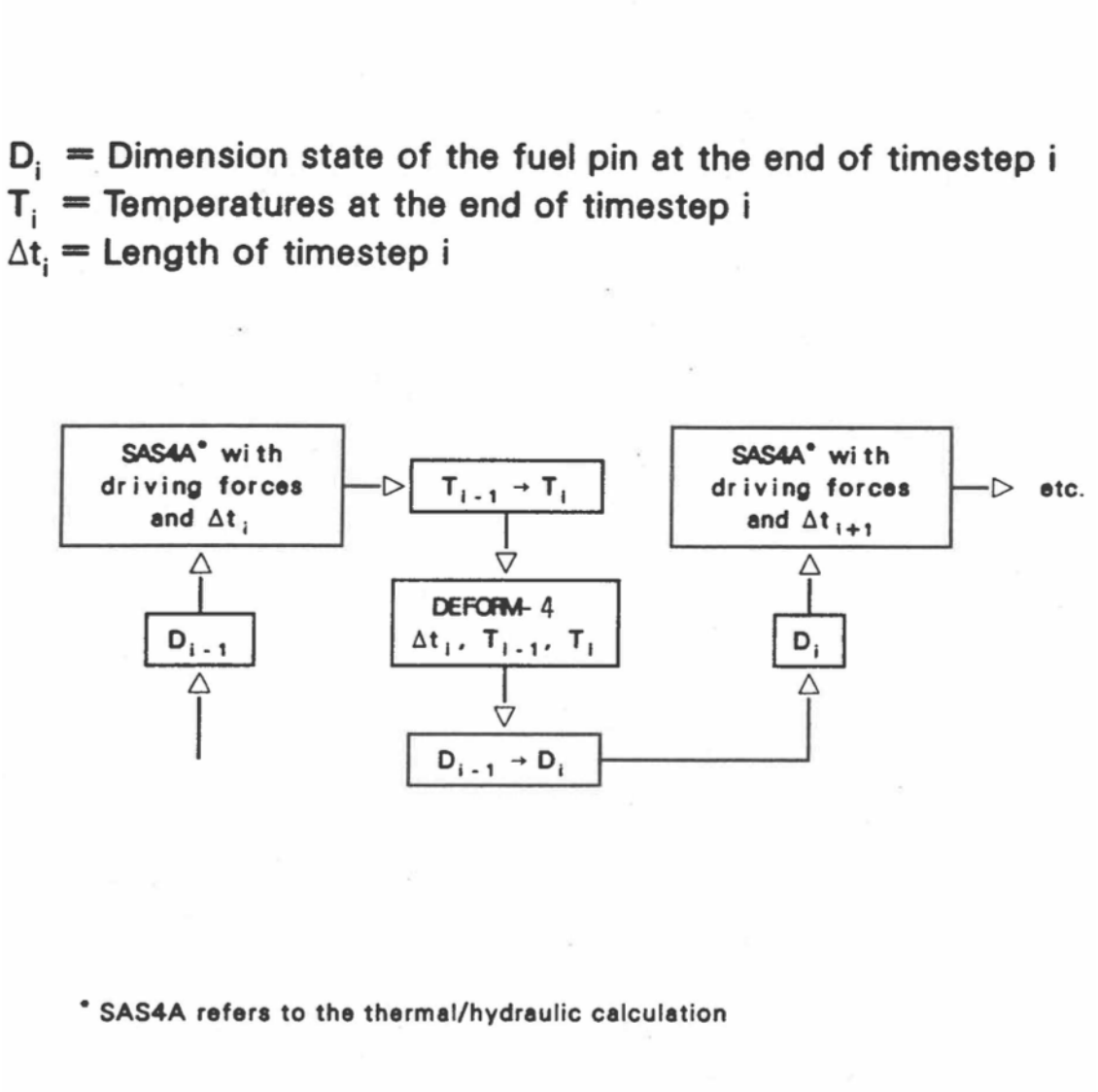


Figure 8.5.1: DEFORM-4 - SAS4A Time Step Interaction Scheme

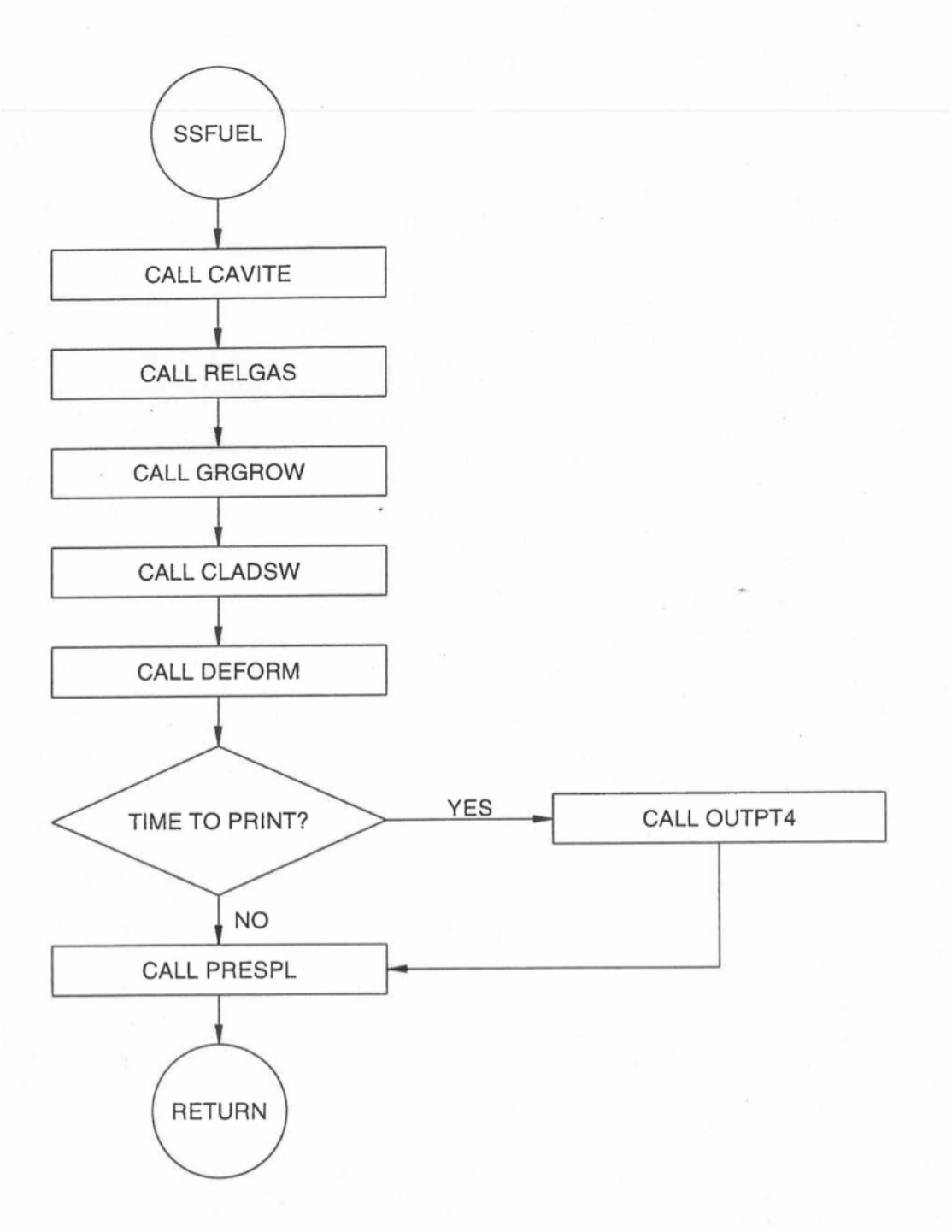


Figure 8.5.2: Pretransient DEFORM-4 Driver (SSFUEL) Flow Chart

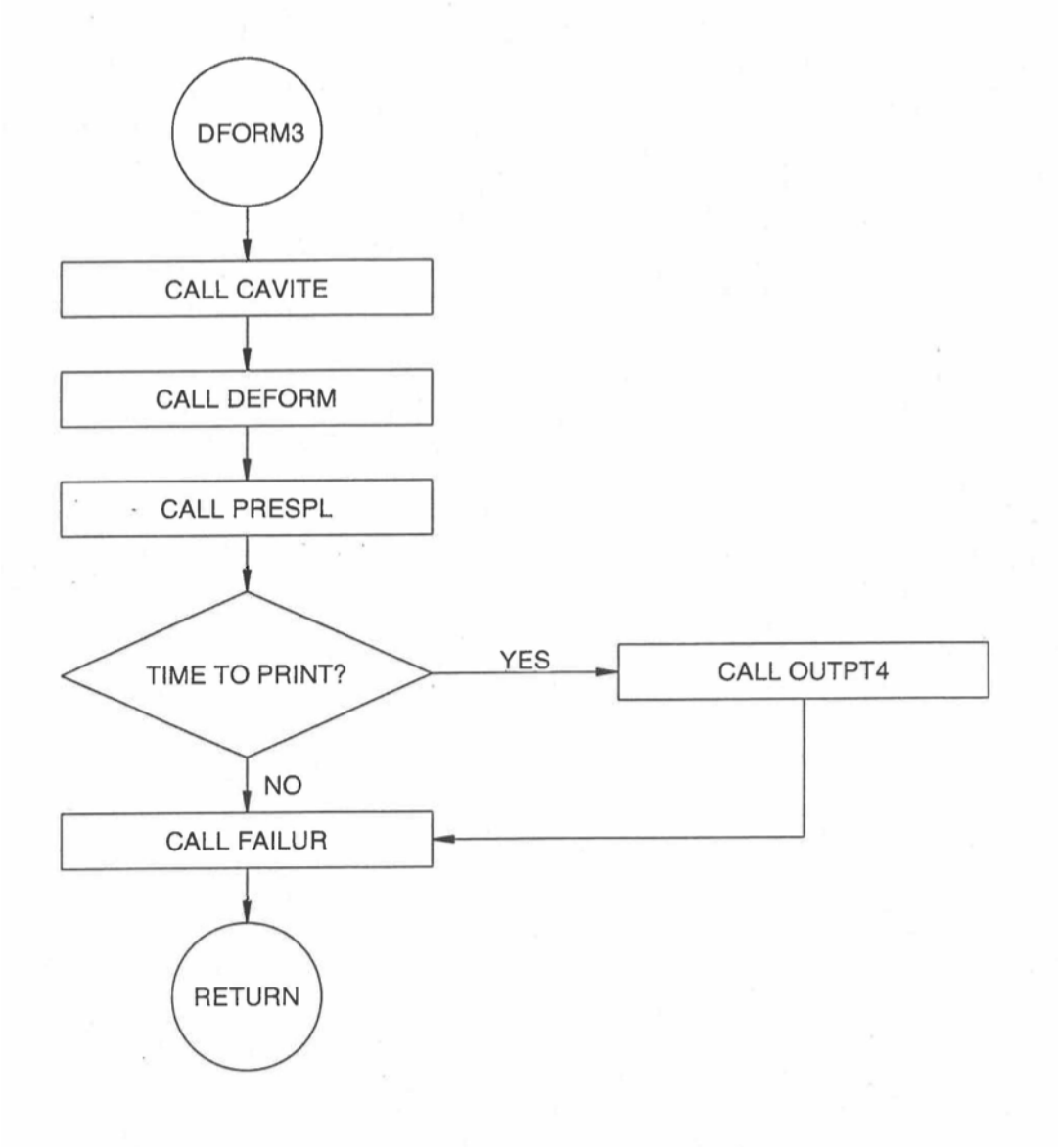


Figure 8.5.3: Transient DEFORM-4 Driver (DFORM3) Flow Chart

8.5.1 Thermal-mechanical Solution

The thermal-hydraulic routines in SAS4A provide DEFORM-4 with the new coolant pressures and temperature in the fuel, cladding, coolant, and structure. DEFORM-4 must then calculate the thermal-mechanical response of the fuel-cladding system to the changes during the time step. With the temperature changes and coolant pressure now from SAS4A, only one additional boundary condition remains as input to the calculation, the central cavity pressure. If no fuel melting has occurred, or the axial segment is not in the axial region considered to be the molten cavity, this pressure is set equal to the plenum pressure existing at the end of the previous time step. If molten fuel is present, then the axial pressure distribution is determined from the results of the molten cavity pressure subroutine CAVITE, see Section 8.3.7. This is the boundary force acting on the inner surface of the non-molten fuel. The remaining force necessary to completely describe the external force system is the fuel-cladding interface pressure, and this is determined through the iterative procedure described below.

In Figure 8.5.4, the main sequence of the mechanics driven routine, DEFORM, is illustrated. After initializing certain parameters, such as the thermal expansion strains, moduli of elasticity, etc., the fuel is checked to see if it is fully cracked or molten out to the cracked region. If a solid, non- cracked annulus of fuel still exists, the solid fuel elastic solution routine SOLID is called. If it is determined in this routine that the fuel becomes completely cracked during the time step, the crack fuel elastic solution routine MKDRIV is called. This routine would have been called initially if the fuel was completely cracked at the beginning of the time step.

The sequence of operations within the subroutine SOLID is shown in Figure 8.5.5. The main purpose of this routine is to determine the correct fuel-cladding interface pressure when a solid, uncracked fuel annulus exists at the beginning of the time step. Since the coolant pressure and central cavity pressure have been determined, SOLID must determine the fuel-cladding interface pressure that produces consistency between the outer fuel surface and inner cladding surface locations.

The approach used in the solution procedure in SOLID is to recognize that the fuel-cladding interface pressure is a bounded variable. It is, at least, the plenum pressure, but cannot exceed the value that would produce plastic flow in the cladding. SOLID therefore iterates to determine the value between these bounds that produces a consistent set of conditions. As a first estimate, the interface pressure from the previous time step is used. This, together with the thermal expansion in the current time step, is used to calculate the axial expansion in EXPAND, the non-molten fuel stresses and strains in FSIGMA, the crack volume fraction in CRAKER, the cladding stresses and strains in CSIGMA, and the fuel swelling strains in FSWELL. Each of these routines calculate the incremental changes that occur from the previous time step results in response to the changes occurring during the current time step. If complete cracking of the fuel occurs during the time step, control is passed to the subroutine MKDRIV. If part of the fuel remains uncracked, then the new fuel-cladding gap is examined to determine if a consistent set of conditions exists.

If a gap exists and the interface pressure is the plenum pressure, the solution is finished. If there is an interface pressure greater than the plenum pressure and the fuel-cladding gap is zero, then the solution is finished. If there is a negative gap and the interface pressure produces the flow stress, the cladding is in plastic deformation and the fuel determines the deformation, so the solution is finished.

The more interesting case involves a negative gap with an interface pressure below that necessary to cause plastic deformation of the cladding. In this case the interface pressure is first set to that which would cause plastic flow, and the calculations are repeated. This second calculation through EXPAND, FSIGMA, CRAKER, CSIGMA, and FSWELL then provides the upper bound on the interface pressure. If the gap becomes positive, the correct value of the interface pressure is between the initial and the new estimate. Based on the changes in gap size as a function of the change in interface pressure, and the magnitude of the deformation necessary to produce a match of the outer fuel and inner cladding surfaces, a third estimate of the interface pressure is obtained assuming a linear relationship between interface pressure and gap size. This procedure is continued until a fuel-cladding surface match is achieved within the desired tolerance or the fuel cracks completely.

When the applied interface pressure, cavity pressure, and thermal stresses produce a stress state in the fuel sufficient to cause the circumferential stress to exceed the fuel crack strength in a previously uncracked fuel cell,

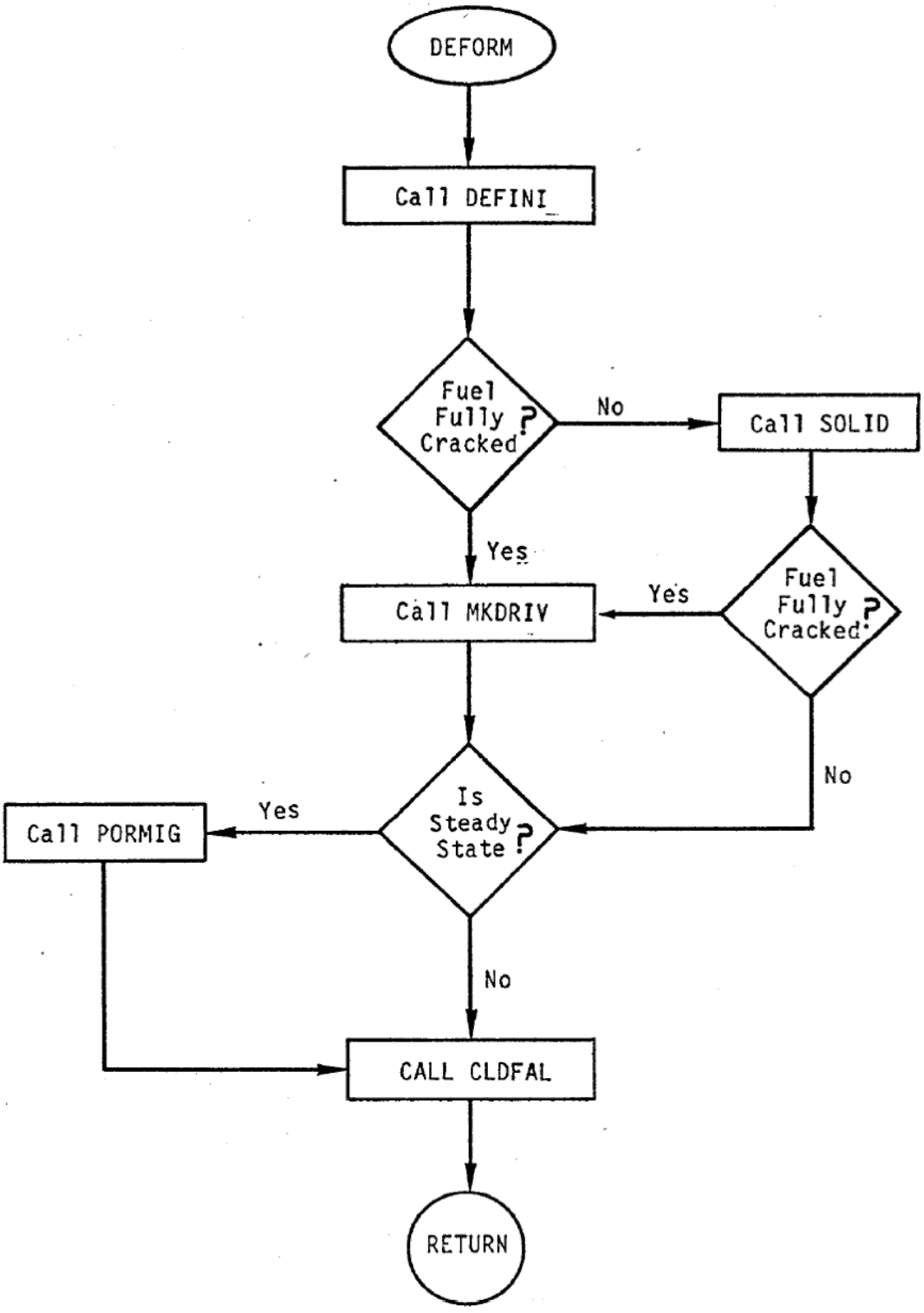


Figure 8.5.4: DEFORM-4 Mechanics driver (DEFORM) Flow Chart

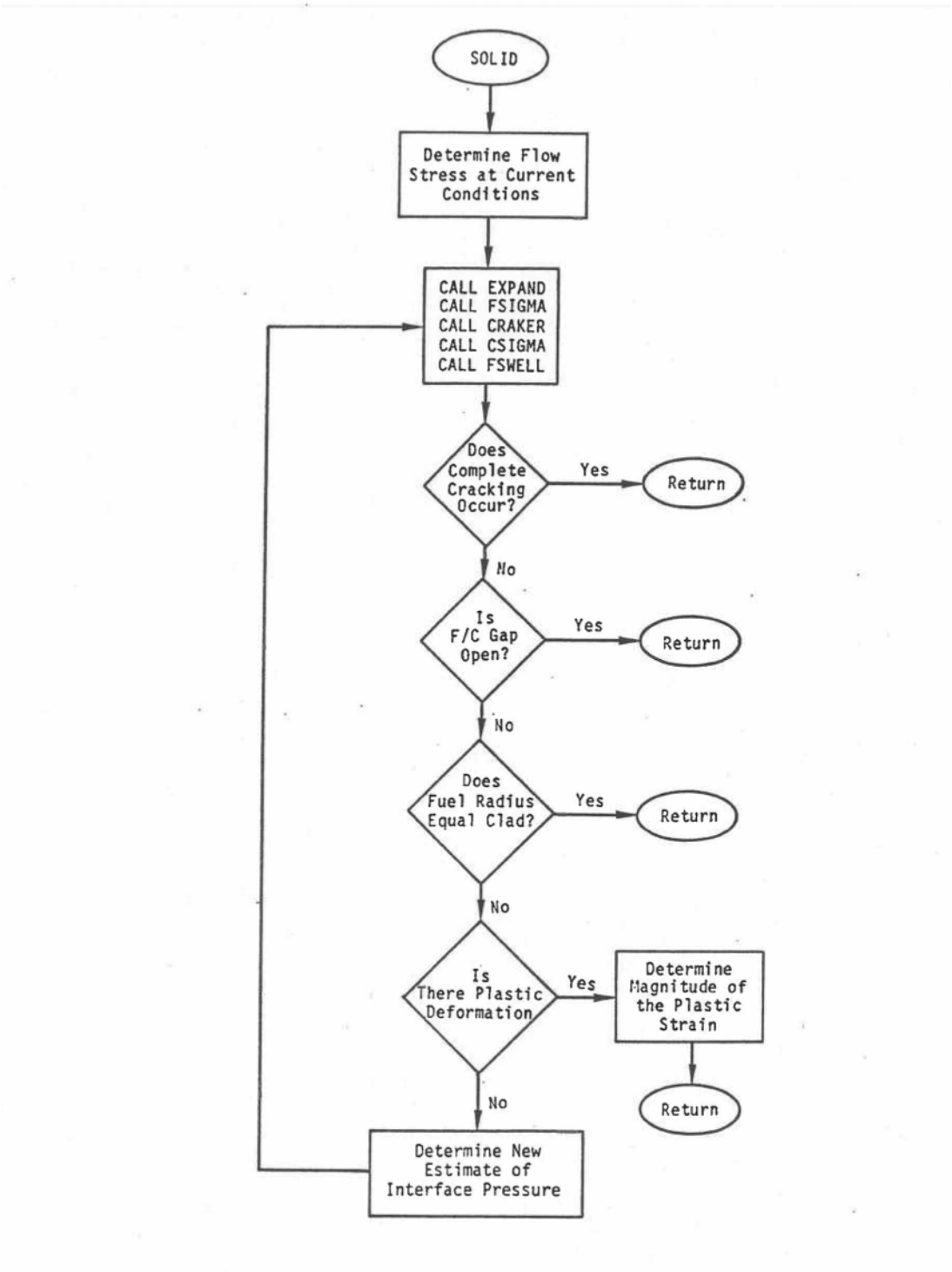


Figure 8.5.5: DEFORM-4 Mechanics Interaction Solution (SOLID) Flow Chart

then the routine FSIGMA will redo its internal calculation with the new distribution of uncracked and cracked fuel cells. This routine will allow cracking to occur in all nodes where the circumferential stress exceeds the fracture strength. Prior to each invocation of EXPAND, FSIGMA, CRAKER, CSIGMA, and FSWELL, the crack state of the fuel is reset to the condition at the beginning of the time step.

In the case where all non-molten fuel is cracked, the thermal-mechanical solution is controlled by the subroutine MKDRIV. The sequence of operations for this routine is shown in Figure 8.5.6. There is no iteration because the fuel-cladding interface pressure must be consistent with the cavity pressure and the solid fuel geometry. Knowing these two boundary conditions, the strains and fission gas induced swelling can be determined. If the fuel-cladding interface is open and the cavity pressure is equal to the plenum pressure, no movement of the cracked fuel takes place beyond the thermal-elastic and swelling strains. If the cavity is pressurized, the cracked fuel is translated radially to the cladding location. If the fuel-cladding gap is negative, the fuel is moved inward to provide a match at the cladding location. In both cases, the crack volumes are adjusted to be consistent with the radial translation.

In the pre-transient calculation, the cracked fuel is allowed to heal if the temperature is high enough. A healing temperature is determined based on the fuel solidus temperature.

$$T_{\mathrm{H}} = f_{\mathrm{H}} T_{\mathrm{s}} \tag{8.5-1}$$

where

T_{H} = Crack healing temperature, K

f_{H} = Input fraction

T_{s} = Fuel solidus temperature, K

If the fuel cell temperature is higher than this healing temperature, the cell is assumed to become solid again. This healing does not take place in the transient calculation where the time scale is too short.

8.5.2 Plastic Fuel Deformation Solution

The routine FSWELL handles all plastic fuel deformations. The basic equations for the non-equilibrium fission-gas bubbles are given in Section 8.3.4.

First, the fuel hydrostatic pressure in each radial fuel node is calculated. The fission-gas bubble swelling or hot pressing in each node is then determined. Once the swelling strains are known, the new fuel node boundaries are calculated.

In determining whether a fuel cell will move radially outward or inward, the concept of a plasticity temperature is used. If the fuel temperature is above the plasticity temperature, T_{sep}, it is assumed that the fuel is soft enough to be easily deformed and will move radially inward in response to the restriction of the cooler fuel. This softening temperature is defined as

$$T_{\mathrm{sep}} = T_{\mathrm{s1}} + T_{\mathrm{s2}} \sigma_{\mathrm{fci}} \tag{8.5-2}$$

where

T_{s1} = Input base softening temperature, K

T_{s2} = Pressure effect constant, K Pa^{-1}

σ_{fci} = Fuel-cladding interface stress, Pa

The cooler fuel cells move radially outward. If the central void closes, the inner radius is fixed to a nominal zero value, 1×10^{-6}, and all movement is radially outward. If the fuel is cracked across its entire radial extent, the movement is determined as described above.

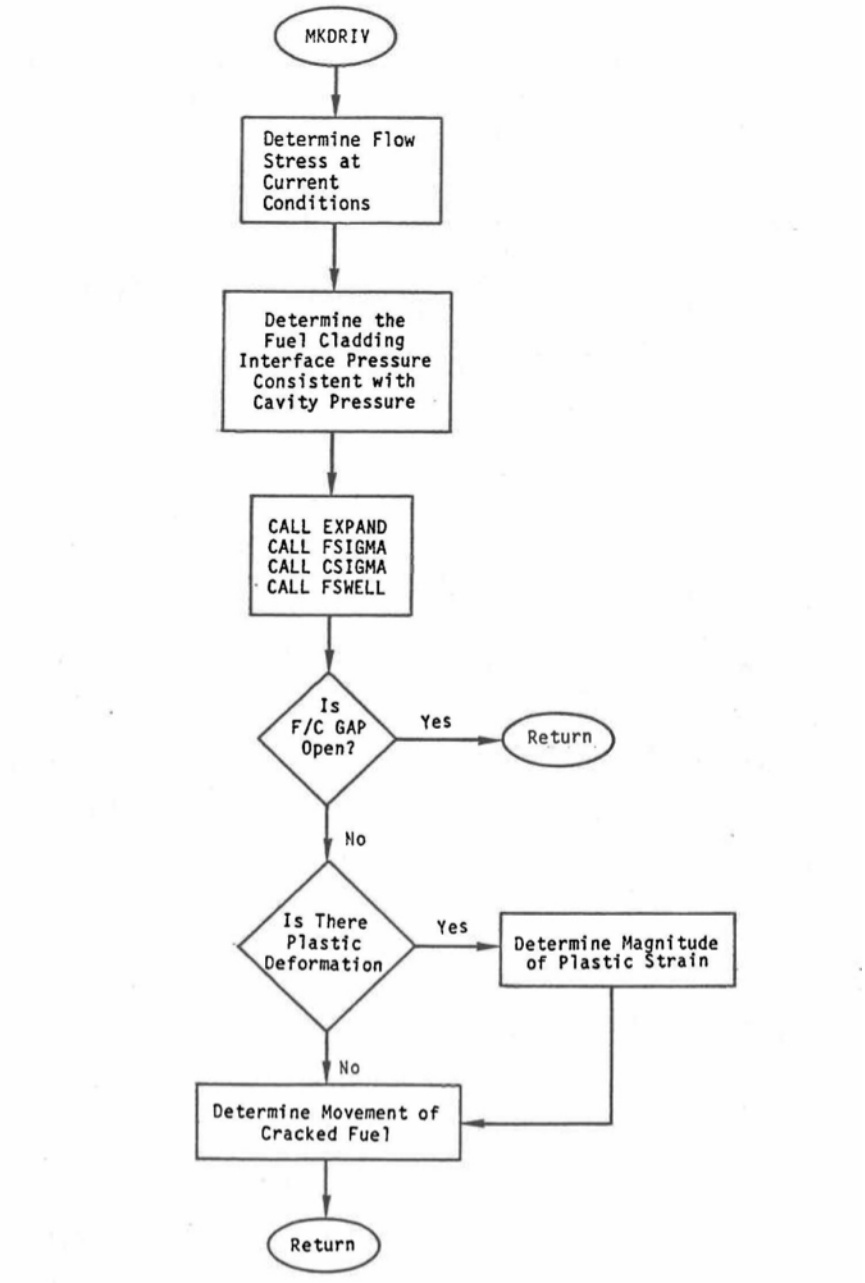


Figure 8.5.6: DEFORM-4 Mechanics Interaction Solution with Fully Cracked Fuel (MKDRIV) Flow Chart

In the molten cavity region, the volume changes calculated by CAVITE (see Section 8.3.7) are included in the redefinition of the node boundaries. In the solid nodes, the remaining as-fabricated porosity and any fuel crack volume are included with the fission-gas porosity in determining the new cell volumes. Once all the new node boundaries have been determined, the variation from the original is calculated and these node deformations are passed to the controlling solution routine.

Once a solution is obtained, all dimensions and stresses are updated to their final values, and the final plastic strain calculated (if any).

8.5.3 Final DEFORM-4 Calculations

After the thermal-mechanical calculations are finished, DEFORM-4 calculates the fuel axial expansion reactivity effects (see Section 8.3.9). The new plenum pressure, based on the new dimensions, temperatures, and fission gas release is then calculated in the subroutine PRESPL. The new gap conductance values for use in the next time step are determined. Control of the calculation is then returned to the SAS4A thermal/hydraulic routines for the calculation of the temperatures and coolant state for the next time step.

8.6 Computer Code Implementation

The DEFORM-4 module is implemented in SAS4A as a group of subroutines. Section 8.6.1 gives a list of these subroutines and a brief description of the calculation performed by each. The direct coupling with SAS4A is through the subroutine SSFUEL for the pre-transient calculation, and DFORM3 for the transient calculation.

The input parameters necessary for DEFORM-4 to perform its calculations are given in Table 8.6.3. This table also includes cross-references to the equation or section number where the parameter is used, the input location and variable name, a suggested value (assuming mixed-oxide fuel with 20% cold-worked, stainless-steel cladding), and the external reference where the suggested value originated. Further information on the input parameters is given in the Input Description (see Chapter 2 Section 2.2).

In Section 8.6.3 a brief description of the output generated by DEFORM-4 is given. The sample output given in Figure 8.6.1 through Figure 8.6.13 are included for illustration only, and should not be used as a sample case.

8.6.1 Subroutine and Function List

Table 8.6.1: DEFORM-4 Associated Subroutines and Functions

Routine	Description
ALPHC	Mean thermal expansion coefficient of the cladding
ALPHF	Mean thermal expansion coefficient of the fuel
CAVITE	Molten cavity gas release and pressurization
CLADSW	Irradiation-induced cladding swelling
CLDFAL	Cladding failure fraction calculations
CRAKER	Volume fraction of cracks in fuel
CSIGMA	Thermal-elastic calculation in the cladding
DEFINI	Initial setup routine
DEFORM	Main mechanics driver routine
DFORM3	DEFORM-4 interface to SAS4A for transient calculation
ECLADF	Cladding modulus of elasticity function
EFUELF	Fuel modulus of elasticity function

continues on next page

Table 8.6.1 – continued from previous page

Routine	Description
EXPAND	Thermal-mechanical axial expansion routine
FAILUR	Determination of reaching failure criterion
FK	Fuel thermal conductivity function
FSIGMA	Thermal-elastic calculation in the fuel
FSWELL	Fission-gas bubble swelling/compaction routine
GRGROW	Grain growth in the fuel
HGAP	Fuel-cladding gap conductance function
KFUEL	Thermal conductivity of the fuel
MKDRIV	Thermal-elastic and interface pressure control routine when solid fuel is completely cracked
NABOND	Fuel-cladding gap conductance for sodium-bonded metal fuel
OUTPT3	Short form output routine
OUTPT4	Standard output routine
PORMIG	As-fabricated porosity migration routine
PRESPL	Plenum pressure calculation
RELAX	Stress relaxation in the fuel (not operational)
RELGAS	Fission gas release in the fuel
RHOF	Theoretical density of fuel, solid and liquid
RHOL	Density of liquid fuel (not currently used)
RHOS	Density of solid fuel (not currently used)
SOLID	Thermal-elastic and interface pressure iteration control routine
SIGFRA	Fracture strengths of the fuel
SSFUEL	DEFORM-4 interface to SAS4A for pre-transient irradiation
UTS	Ultimate tensile strength of the cladding
YLDCF	Yield point of the cladding

Table 8.6.2: DEFORM-4 Subroutines and Functions Associated with 15-15Ti Cladding

Routine	Description
InitCheck-Clad15Ti (cladthickness, NCHAN)	Checks for the errors and calls INITSCLAD15Ti in order to allocate the required derived data type for each channel. “cladthickness” is the thickness of the cladding and NCHAN is the total number of channels.
INITSCLAD15Ti (ICH)	Initializes a CLAD15 derived typed for channel ICH. Called from SSFUEL.
BEGIN-CLAD15Ti (j)	Used to preserve initial values of the variables at each time step. Sets the values of the target variables using dummy storage variables for each axial node j.
ENDCLAD15Ti (j)	Used to update the dummy storage variables using the computed values of the target variables at the end of each time step for each axial node j.
CDFNORMALD-IST (j)	Computes the clad failure fraction as a function of logarithm of CDF at axial location j.
INPT_CLAD15Ti ReadDat (unit,ich)	Reads the relevant variables for the 1515Ti cladding for channel “ich” from the restart file with an assigned file number, “unit”. It also calls INITSCLAD15Ti in order to allocate the required derived data type for each channel.
INPT_CLAD15Ti WriteDat (unit,ich)	Writes the relevant variables for the 1515Ti cladding for the channel “ich” to the restart file with an assigned file number, “unit”.
ET15TI (sigma, temp)	Computes clad thermal creep rate for a given applied stress (sigma) and temperature (temp).
EPSWC15TI (neutronflux, fluence, dosconversion, temp)	Computes and returns the linear clad irradiation induced void swelling rate for 15-15Ti cladding, given the fast neutron flux (neutronflux), fast neutron fluence (fluence), dose conversion factor (doseconversion) and temperature (temp).
EPSIR15TI (temp, neutronflux, stress, dosconversion)	Computes and returns the clad irradiation creep rate for 15-15Ti cladding, given the temperature (temp), fast neutron flux (neutronflux), applied stress (stress) and dose conversion factor (dosconversion).
CRUP15TI (sigma, temp, dose)	Computes the creep rupture time of the cladding given the applied stress (sigma), temperature (temp), and cladding dose (dose).

8.6.2 Input Variables

Table 8.6.3: DEFORM-4 Input Variables

Equation Variable	Reference Eq. Or Section No.	SAS4A Input Variable	Block	Location	Suggested Value	External Reference
-	-	**IPROPT**	1	6	0	-
-	-	**IYLD**	1	26	2	-
-	8.3.7	**IMELTV**	1	28	1	-
-	-	**IXSTPC**	1	32	0	-
-	-	**IXSTPF**	1	33	0	-
-	-	**IMCVTY**	1	34	0	-
-	-	**IAREXT**	1	40	0	-

continues on next page

Table 8.6.3 – continued from previous page

-	-	**POW**	12	1	-	-
-	Eq. (8.3-166)	**BETADN**	12	4-9	-	-
ρ_o	Eq. (8.7-1)	**COEFDS**(1)	13	1	11.08D+3	8-17
C_1	Eq. (8.7-1)	**COEFDS**(2)	13	2	2.04D-5	8-17
C_2	Eq. (8.7-1)	**COEFDS**(3)	13	3	8.70D-9	8-17
C_1	Eq. (8.7-13)	**COEFK**(1)	13	4	2.10D+0	8-32
C_2	Eq. (8.7-13)	**COEFK**(2)	13	5	2.88D-3	8-32
C_3	Eq. (8.7-13)	**COEFK**(3)	13	6	2.52D-5	8-32
C_4	Eq. (8.7-13)	**COEFK**(4)	13	7	5.83D-10	8-32
C_5	Eq. (8.7-14)	**COEFK**(5)	13	8	5.75D-2	8-32
C_6	Eq. (8.7-14)	**COEFK**(6)	13	9	5.03D-4	8-32
C_7	Eq. (8.7-14)	**COEFK**(7)	13	10	2.91D-11	8-32
T_r	Eq. (8.2-5)	**TR**	13	419	300.	-
MW_{fg}	Eq. (8.3-24)	**FGMM**	13	600	131.	-
f_g	Eq. (8.3-22)	**GATPF**	13	601	0.246	8-17
E_f	Eq. (8.3-23)	**ENPF**	13	602	197.	-
R_{eq}	Eq. (8.3-7)	**RLEQ**	13	603	0.6	-
R_{ueq}	Eq. (8.3-21)	**RUEQ**	13	604	1.2	-
-	8.3.1	**PRSMIN**	13	605	0.02	-
T_s	Eq. (8.3-87)	**TFSOL**	13	786-793	-	-
T_l	Eq. (8.3-87)	**TFLIQ**	13	794-801	-	-
P_o	Eq. (8.3-7)	**PRSTY**	13	1073-1080	-	-
Not currently used	-	**AC**	13	1081	-	-
Not currently used	-	**QSWL**	13	1082	-	-
A_p	Eq. (8.3-1)	**APORE**	13	1083	20.704	8-6
Q_p	Eq. (8.3-1)	**QPORE**	13	1084	4.5281D+5	8-6
A	Eq. (8.3-1)	**ABC**	13	1085	1.5	8-6
R	-	**RGASSI**	13	1086	8.31433	-
γ	Eq. (8.3-45)	**GAMMA**	13	1087	0.45	8-6
A_{pg}	Eq. (8.3-44)	**APG**	13	1088	5.0D+4	8-6
Q_{pg}	Eq. (8.3-44)	**QPG**	13	1089	56506.5	8-6
G_k	Eq. (8.3-11)	**GK**	13	1090	1.717D+10	8-11
Q_v	Eq. (8.3-11)	**QV**	13	1091	3.87D+5	8-11
G	Eq. (8.3-13)	**GK1**	13	1092	1.4556D-8	8-12
Q	Eq. (8.3-13)	**QV1**	13	1093	2.67D+5	8-12
G_m	Eq. (8.3-12)	**GRAINK**	13	1094	2.23D-3	8-12
Q_m	Eq. (8.3-12)	**GRAINQ**	13	1095	6.3375D+4	8-12
C_v	Eq. (8.3-145)	**CVXE**	13	1096	94.69	8-30
C_v	Eq. (8.3-145)	**CVHE**	13	1097	3.13D+3	8-30
δ_f	Eq. (8.3-143)	**ROFF**	13	1098	3.30D-06	-

continues on next page

Table 8.6.3 – continued from previous page

δ_c	Eq. (8.3-143)	ROFC	13	1099	1.78D-06	-
Not currently used	-	AZEROX	13	1100	-	-
γ	Eq. (8.3-145)	GAMGS	13	1101	1.66	8-30
Not currently used	-	HARDNS	13	1102	-	-
Not currently used	-	ET	13	1103	1.0	8-30
a_i	Eq. (8.3-144)	ACCHE	13	1104	0.15	8-19
a_i	Eq. (8.3-144)	ACCXE	13	1105	0.85	8-19
C_o	Eq. (8.3-146)	CZERO	13	1106	1.98	8-30
σ	Eq. (8.3-135)	STEBOL	13	1107	5.69D-8	-
ε_f	Eq. (8.3-135)	EMSF	13	1108	0.9	8-19
ε_c	Eq. (8.3-135)	EMSC	13	1109	0.8	8-19
Q_{A1}	Eq. (8.3-42)	QA1	13	1110	6.92D+3	8-16
Q_{A2}	Eq. (8.3-42)	QA2	13	1111	33.95	8-16
Q_{A3}	Eq. (8.3-43)	QA3	13	1112	0.338	8-16
Q_{A4}	Eq. (8.3-43)	QA4	13	1113	1.48D+4	8-16
Q_{A5}	Eq. (8.3-43)	QA5	13	1114	9.575	8-16
A_α	Eq. (8.3-33)	ALFSS	13	1115	2.0D-04	-
Q_α/R	Eq. (8.3-33)	BETSS	13	1116	1.1D+04	-
v_c	8.2	CNU	13	1117	0.3	8-17
v_f	8.2	FNU	13	1118	0.3265	8-17
Not currently used	-	AM	13	1119	-	-
Not currently used	-	QLAX	13	1120	-	-
Not currently used	-	QLAX2	13	1121	-	-
Not currently used	-	DDX	13	1122	-	-
Not currently used	-	DDX2	13	1123	-	-
f_{gb}	Eq. (8.3-29)	FIFNGB	13	1148	0.10	-
f_m	Eq. (8.3-81)	FNMELT	13	1169	0.30	-
τ_g	Eq. (8.3-107)	CIRTFS	13	1170	16.67	-
C_3	Eq. (8.7-2)	COEFDL(2)	13	1201	9.3D-5	8-17
Not currently used	-	QSTAR	13	1202	-	-
Not currently used	-	ABCPU	13	1203	-	-

continues on next page

Table 8.6.3 – continued from previous page

Not currently used	-	QPU	13	1204	-	-
Not currently used	-	DPUO	13	1205	-	-
Not currently used	-	TMIDFG	13	1218	-	-
Not currently used	-	FGSPRD	13	1219	-	-
Not currently used	-	FGPORX	13	1220	-	-
Not currently used	-	FGMIN	13	1221	-	-
Not currently used	-	HECOND	13	1222	1.58D-3	8-38
Not currenty used	-	FGCOND	13	1223	7.2D-5	8-38
Not currently used	-	AKCOND	13	1224	0.75	8-38
M_n	Eq. (8.3-140)	HEMM	13	1225	4	-
$_{sfp}$	Eq. (8.3-62)	EPSSFP	13	1226	2.46D-3	8-17
Not currently input	-	HEMASX	13	1227	-	-
Not currently input	-	ZSWFAC	13	1228	-	-
A_f	Eq. (8.2-57)	FAXIAL	13	1258	1.0	-
ε_{cw}	Eq. (8.3-62)	FCLDWK	13	1259	-	-
Not currently used	-	FMELTD	13	1260	-	-
Not currently used	-	FSTRAN	13	1261	-	-
f_H	Eq. (8.5-1)	FTMPCH	13	1262	0.6	-
f_a	Eq. (8.3-116)	EXPCOF	13	1263	1.0	-
-	-	PRTSTR	13	1264	0.0	-
-	-	PRTDEL	13	1265	0.0	-
-	8.4.3	FIRLIM	13	1266	0.80	-
-	8.4.3	SECLIM	13	1267	0.95	-
-	8.4.3	THRLIM	13	1268	0.98	-
-	8.4.3	DTFAL1	13	1269	0.020	-
-	8.4.3	DTFAL2	13	1270	0.010	-
-	8.4.3	DTFAL3	13	1271	0.002	-
Not currently used	-	AKD	13	1272	-	-
Not currently used	-	CKD	13	1273	-	-
Not currently used	-	QKD	13	1274	-	-
-	8.3.6	FGFI	13	1275	-	-
-	8.7.4	IRHOK	51	3	3	-

continues on next page

Table 8.6.3 – continued from previous page

N	Eq. (8.3-27)	NT	51	14	11	-
-	-	IFUELV	51	15	-	-
-	-	IFUELB	51	16	-	-
-	-	ICLADV	51	17	-	-
-	8.3.3	KTING	51	19	0	-
-	-	NAXOP	51	20	25	-
-	-	MSTEP	51	21	-	-
-	8.3.5	ITAU	51	22	0	8-17
-	8.3.5	IRATE	51	23	0	8-17
-	8.3.8	IHGAP	51	24	1	-
N_{pps}	Eq. (8.3-166)	NPIN	51	25	-	-
N_{spc}	Eq. (8.3-166)	NSUBAS	51	26	-	-
-	8.1	MZUB	51	27	-	-
-	8.1	MZLB	51	28	-	-
-	-	IRELAX	51	30	0	-
n	Eq. (8.3-11)	NGRAIN	51	31	4	8-11
-	-	ISSFUE	51	32	1	-
-	-	NPLIN	51	37-44	-	-
-	-	JPRNT1	51	46	0	-
-	-	JPRNT2	51	47	0	-
-	-	NNBUG1	51	48	0	-
-	-	NNBUG2	51	49	0	-
-	-	IDBUGF	51	50	0	-
-	-	NSKIP	51	51-58	-	-
-	-	MFAIL	51	86	3	-
-	-	IFAIL	51	87	0	-
-	-	JFAIL	51	88	0	-
-	8.3.4.1	IPSIG	51	95	3	-
-	-	IHTPRS	51	96	0	-
-	-	IPRD	51	97	3	-
-	8.1	IEQMAS	51	118	1	-
-	-	ISSFU2	51	122	1	-
-	-	IHEALC	51	123	2	-
-	-	IAXTHF	51	124	1	-
A	8.3.9	AXHI	61	8-31	-	-
-	-	PLENL	61	53	-	-
-	-	RBR	61	54-77	-	-
-	-	RER	61	78-101	-	-
-	-	RBRPL	61	102	-	-
-	-	RERPL	61	103	-	-
-	-	RINFP	61	104-127	-	-
-	-	ROUTFP	61	128-151	-	-
-	-	RBR0	61	180	-	-
-	-	RER0	61	181	-	-
-	-	PSHAPE	62	6-29	-	-
S	Eq. (8.3-27)	PSHAPR	62	30-44	-	-
-	-	PLIN	62	45-52	-	-
-	-	TPLIN	62	53-60	-	-

continues on next page

Table 8.6.3 – continued from previous page

-	-	FLTPOW	62	61	-	-
-	8.3.9	FUELRA	62	208-231	-	-
A	Eq. (8.3-152)	AHBPAR	63	2	0.0	8-22
B	Eq. (8.3-152)	BHBPAR	63	3	0.0	8-22
C	Eq. (8.3-152)	CHBPAR	63	4	0.0	8-22
-	8.3.8	HBMAX	63	5	4.0D+4	8-22
-	8.3.8	HBMIN	63	6	1.0D+2	-
H	Eq. (8.3-152), Eq. (8.3-153)	HBPAR	63	7	0.0	-
D_o	Eq. (8.3-21)	DGO	63	26	1.0D-5	-
P_{ogas}	Eq. (8.3-6)	P0GAS	63	27	-	-
	-	FCLOP	63	68	-	-
T_{s1}	Eq. (8.5-2)	TSEP1	63	69	2.2D+3	-
T_{s2}	Eq. (8.5-2)	TSEP2	63	70	1.0D-8	-
	-	BONDNA	63	71	-	-
	-	REFDEN	63	72	-	-
-	8.4	FSPEC	65	1	-	-
-	8.4	FMELTM	65	2	0.20	-
		IDEFOPT	INPCHN	497	-	-
		ICTYPE	INPCHN	225	-	-
	Eq. (8.7-52)	IDEFST-FAL	INPCHN	498	-	-
	Eq. (8.7-49)	CDOSEC-ONV	PMATCH	149	-	-
	Section 8.7.17.5	COCOR	PMATCH	150	-	-
μ	Eq. (8.7-52)	CDFMEAN	PMATCH	151	-	-
σ	Eq. (8.7-52)	CDFSIGMA	PMATCH	152	-	-

8.6.3 Output Specific to DEFORM-4

Figure 8.6.1 through Figure 8.6.13 show a sample of the output generated by DEFORM-4. This sample is for a section of pre-transient output, but the same output is generated in the transient case. What follows is a very brief discussion of the output format for each page.

The standard SAS4A coolant temperature/pressure summary.

Fuel node boundary locations and cell temperatures. Radial and axial node numbers are shown. Under "IZ" is the node number of the first non-molten cell. Under "IETA" is the node number of the last solid, i.e., un-cracked, cell. Therefore, IZ to IETA defines the extent of the solid annular fuel region.

Fuel/cladding summary information is presented. The titles are self-explanatory except for the third one in the bottom section. This column shows the ratio of the calculated gap conductance to that used for the next time step thermal calculation. Since there is no iteration between the DEFORM-4 mechanics calculation and the SAS4A thermal calculation, a scheme has been employed to avoid gross oscillations in the gap conductance and temperatures. This is to use the average of the calculated and previously used gap conductance for the next time step. In order to determine how close the values are to those calculated, this ratio is printed.

CHANNEL 1 SAS4A 3.0 DEFORM-4 TEST PROBLEM FOR SAS4A/SASSYS-1 VERSION 3.0 01/ 9/94 18:20:20 PAGE 114
JOB: 21122/pepper ARGONNE NATIONAL LABORATORY, JANUARY, 1994 USER: 22157/cahalan

*** STEADY STATE ***

FLOWRATE (KG/S) 0.1336690

COOLANT PROFILE

AXIAL LOCATION (M)	COOLANT TEMPERATURE (K)	SATURATION TEMPERATURE (K)	PRESSURE (MPA)
VESSEL OUTLET			0.1526
2.99720	817.973	1206.415	0.1526
2.78130	817.973	1231.720	0.1873
2.56540	817.973	1253.546	0.2221
2.34950	817.973	1272.830	0.2568
2.13360	817.973	1290.166	0.2916
1.93040	817.973	1305.067	0.3243
1.72720	817.973	1318.840	0.3570
1.64254	817.693	1324.617	0.3714
1.55787	817.245	1330.223	0.3859
1.47320	816.527	1335.671	0.4003
1.42240	815.604	1338.869	0.4090
1.37160	814.378	1342.017	0.4177
1.30387	806.591	1346.168	0.4294
1.23614	797.065	1350.239	0.4411
1.16840	785.948	1354.263	0.4529
1.10067	773.836	1358.212	0.4647
1.03294	760.735	1362.087	0.4765
0.96520	746.886	1365.887	0.4883
0.89747	732.732	1369.613	0.5001
0.82973	718.434	1373.270	0.5119
0.76200	704.246	1376.862	0.5237
0.69427	690.342	1380.390	0.5354
0.62653	677.144	1383.853	0.5472
0.55880	664.883	1387.255	0.5589
0.49107	653.529	1390.597	0.5706
0.42333	643.629	1393.878	0.5823
0.35560	635.284	1397.102	0.5940
0.30480	633.788	1399.472	0.6027
0.25400	632.586	1401.816	0.6113
0.12700	631.055	1407.562	0.6330
0.00000	630.150	1413.159	0.6547
-0.16933	630.150	1420.156	0.6826
-0.33867	630.150	1426.940	0.7105
-0.50800	630.150	1433.525	0.7384
-0.67733	630.150	1439.924	0.7663
-0.84667	630.150	1446.151	0.7942
-1.01600	630.150	1452.214	0.8221
-1.24460	630.150	1460.159	0.8597
-1.47320	630.150	1467.847	0.8974
-1.70180	630.150	1475.297	0.9351
VESSEL INLET			0.9351

Figure 8.6.1: SAS4A Steady-state Coolant Condition Output

CHANNEL 1 SAS4A 3.0 DEFORM-4 TEST PROBLEM FOR SAS4A/SASSYS-1 VERSION 3.0 01/ 9/94 18:20:20 PAGE 115
JOB: 21122/pepper ARGONNE NATIONAL LABORATORY, JANUARY, 1994 USER: 22157/cahalan

AXIAL LOCATION (M)	RADIAL FUEL TEMPERATURE MESH (K) 1	2	3	4	5	6	7	8	9	10	11
1.68487	919.938	918.387	916.838	915.292	913.748	912.207	910.667	909.130	907.595	906.063	904.533
1.60020	974.584	971.982	969.385	966.795	964.211	961.632	959.060	956.494	953.934	951.380	948.832
1.51554	1056.018	1051.545	1047.089	1042.649	1038.225	1033.817	1029.426	1025.051	1020.692	1016.349	1012.023
1.44780	1268.195	1257.046	1245.973	1234.975	1224.049	1213.190	1202.400	1191.689	1181.058	1170.507	1160.039
1.39700	1378.112	1362.331	1346.674	1331.131	1315.664	1300.284	1285.036	1269.945	1255.042	1240.306	1225.709
1.33774	1992.826	1953.113	1895.702	1833.187	1765.621	1690.942	1608.179	1521.167	1435.302	1353.289	1275.610
1.27000	1897.194	1856.771	1793.062	1720.878	1643.399	1556.907	1459.309	1357.685	1260.613	1170.430	1087.026
1.20227	2000.403	1954.640	1881.257	1797.141	1707.806	1608.194	1492.694	1371.410	1257.044	1152.386	1057.565
1.13454	2072.057	2022.771	1944.006	1853.988	1758.661	1653.148	1527.902	1393.572	1266.755	1151.276	1047.883
1.06680	2128.644	2076.085	1991.918	1895.951	1794.430	1684.888	1553.904	1408.759	1270.104	1144.294	1032.849
0.99907	2154.229	2099.197	2010.880	1910.265	1803.921	1691.239	1557.123	1405.786	1259.788	1127.792	1011.713
0.93133	2152.705	2096.253	2005.882	1903.118	1794.671	1679.878	1543.230	1389.485	1241.894	1109.036	992.606
0.86360	2145.721	2088.237	1996.582	1892.577	1782.999	1666.373	1527.196	1372.031	1224.462	1092.228	976.611
0.79587	2124.080	2065.937	1974.092	1870.306	1761.253	1642.634	1500.723	1346.671	1202.545	1073.893	961.359
0.72813	2092.632	2034.383	1943.352	1840.952	1733.586	1612.910	1469.283	1319.012	1180.377	1056.930	948.659
0.66040	2038.440	1981.046	1892.933	1794.371	1689.212	1565.772	1423.945	1283.620	1155.522	1041.365	940.428
0.59267	1957.616	1902.173	1818.608	1725.203	1621.401	1497.796	1364.154	1238.188	1123.800	1021.297	929.573
0.52493	1870.819	1816.410	1736.540	1646.934	1544.664	1426.674	1305.507	1193.335	1091.446	999.406	916.173
0.45720	1773.812	1717.565	1641.100	1556.645	1461.874	1358.456	1256.271	1161.448	1074.356	994.508	921.355
0.38947	1861.437	1796.060	1720.881	1640.102	1551.913	1459.418	1368.681	1282.924	1202.534	1127.278	1057.094
0.33020	1353.173	1334.003	1315.031	1296.250	1277.643	1259.169	1240.870	1222.822	1205.015	1187.433	1170.077
0.27940	1252.789	1238.265	1223.874	1209.615	1195.484	1181.479	1167.590	1153.827	1140.209	1126.740	1113.414
0.19050	1006.549	1000.387	994.258	988.163	982.099	976.069	970.070	964.105	958.171	952.270	946.400
0.06350	875.822	872.593	869.374	866.165	862.967	859.779	856.602	853.435	850.278	847.132	843.996

FUEL NODE	RADIAL FUEL MESH (CM) 1	2	3	4	5	6	7	8	9	10	11	12	IZ	IETA
24	0.0001	0.0671	0.1162	0.1500	0.1775	0.2013	0.2225	0.2419	0.2598	0.2766	0.2924	0.3000	1	12
23	0.0001	0.0671	0.1163	0.1501	0.1776	0.2014	0.2226	0.2420	0.2600	0.2768	0.2926	0.3002	1	12
22	0.0001	0.0672	0.1164	0.1503	0.1778	0.2016	0.2229	0.2423	0.2602	0.2770	0.2929	0.3005	1	12
21	0.0001	0.0674	0.1168	0.1507	0.1783	0.2022	0.2235	0.2430	0.2610	0.2779	0.2938	0.3014	1	11
20	0.0001	0.0675	0.1170	0.1510	0.1787	0.2026	0.2240	0.2435	0.2616	0.2785	0.2944	0.3020	1	8
19	0.0624	0.0907	0.1304	0.1602	0.1854	0.2080	0.2290	0.2488	0.2672	0.2844	0.3006	0.3083	1	2
18	0.0689	0.0947	0.1321	0.1614	0.1863	0.2086	0.2296	0.2495	0.2679	0.2851	0.3013	0.3090	1	2
17	0.0711	0.0964	0.1332	0.1624	0.1871	0.2091	0.2300	0.2499	0.2684	0.2857	0.3018	0.3095	1	2
16	0.0722	0.0973	0.1340	0.1628	0.1873	0.2091	0.2298	0.2497	0.2683	0.2856	0.3018	0.3095	1	2
15	0.0733	0.0982	0.1347	0.1634	0.1877	0.2093	0.2296	0.2495	0.2681	0.2855	0.3017	0.3095	1	2
14	0.0740	0.0988	0.1352	0.1637	0.1880	0.2095	0.2295	0.2493	0.2679	0.2853	0.3016	0.3093	1	2
13	0.0738	0.0986	0.1351	0.1636	0.1879	0.2094	0.2294	0.2492	0.2678	0.2852	0.3014	0.3092	1	2
12	0.0733	0.0983	0.1348	0.1634	0.1876	0.2092	0.2293	0.2491	0.2677	0.2851	0.3014	0.3091	1	2
11	0.0721	0.0973	0.1341	0.1628	0.1871	0.2087	0.2291	0.2491	0.2677	0.2851	0.3013	0.3090	1	2
10	0.0707	0.0963	0.1333	0.1621	0.1865	0.2082	0.2290	0.2491	0.2677	0.2850	0.3011	0.3088	1	2
9	0.0681	0.0944	0.1319	0.1609	0.1855	0.2076	0.2290	0.2491	0.2676	0.2849	0.3010	0.3087	1	2
8	0.0651	0.0921	0.1301	0.1595	0.1845	0.2074	0.2292	0.2492	0.2677	0.2848	0.3009	0.3085	1	2
7	0.0615	0.0895	0.1285	0.1584	0.1839	0.2074	0.2293	0.2493	0.2677	0.2847	0.3007	0.3084	1	2
6	0.0528	0.0844	0.1259	0.1570	0.1834	0.2074	0.2293	0.2492	0.2675	0.2845	0.3005	0.3082	1	2
5	0.0397	0.0788	0.1233	0.1555	0.1827	0.2069	0.2287	0.2486	0.2669	0.2840	0.3000	0.3077	1	2
4	0.0001	0.0675	0.1170	0.1510	0.1787	0.2026	0.2240	0.2435	0.2616	0.2785	0.2944	0.3020	1	7
3	0.0001	0.0674	0.1168	0.1508	0.1784	0.2023	0.2236	0.2431	0.2611	0.2779	0.2938	0.3014	1	9
2	0.0001	0.0672	0.1164	0.1502	0.1778	0.2015	0.2228	0.2422	0.2602	0.2770	0.2928	0.3004	1	12
1	0.0001	0.0671	0.1162	0.1500	0.1775	0.2012	0.2225	0.2418	0.2598	0.2765	0.2923	0.2999	1	12

Figure 8.6.2: SAS4A Fuel Temperature and Node Location Map

CHANNEL 1 SAS4A 3.0 DEFORM-4 TEST PROBLEM FOR SAS4A/SASSYS-1 VERSION 3.0 01/ 9/94 18:20:20 PAGE 117
JOB: 21122/pepper ARGONNE NATIONAL LABORATORY, JANUARY, 1994 USER: 22157/cahalan

AXIAL LOCATION (M)	CLAD TEMPERATURES (K) INNER	MIDPOINT	OUTER	AVERAGE COOLANT TEMPERATURE (K)	STRUCTURE TEMPERATURE (K) INNER	OUTER	AVERAGE FUEL TEMPERATURE (K)	RADIAL LOCATION (CM) INNER FUEL	OUTER FUEL	INNER CLAD	OUTER CLAD
1.685	818.549	818.267	818.006	817.833	817.903	817.962	912.216	0.0001	0.3000	0.3090	0.3654
1.600	818.612	818.161	817.744	817.469	817.580	817.675	961.658	0.0001	0.3002	0.3090	0.3654
1.516	818.719	817.996	817.328	816.886	817.065	817.216	1033.886	0.0001	0.3005	0.3090	0.3654
1.448	819.992	818.444	817.011	816.066	816.448	816.772	1213.500	0.0001	0.3014	0.3090	0.3654
1.397	820.206	818.149	816.246	814.991	815.498	815.929	1300.832	0.0001	0.3020	0.3090	0.3654
1.338	835.290	825.508	816.447	810.484	812.900	814.954	1669.072	0.0624	0.3083	0.3091	0.3655
1.270	832.169	820.193	809.095	801.828	804.782	807.295	1531.116	0.0689	0.3090	0.3090	0.3655
1.202	826.827	812.870	799.930	791.506	794.953	797.887	1575.156	0.0711	0.3095	0.3095	0.3659
1.135	818.376	803.146	789.023	779.892	783.645	786.843	1613.205	0.0722	0.3095	0.3095	0.3658
1.067	808.933	792.423	777.111	767.285	771.344	774.805	1640.108	0.0733	0.3095	0.3095	0.3658
0.999	797.885	780.382	764.144	753.811	758.100	761.761	1644.896	0.0740	0.3093	0.3093	0.3656
0.931	784.919	766.971	750.315	739.809	744.192	747.937	1633.610	0.0738	0.3092	0.3092	0.3655
0.864	771.211	753.021	736.137	725.583	730.009	733.796	1620.385	0.0733	0.3091	0.3091	0.3654
0.796	756.686	738.571	721.755	711.340	715.732	719.492	1598.077	0.0721	0.3090	0.3090	0.3653
0.728	741.813	723.991	707.449	697.294	701.598	705.287	1571.143	0.0707	0.3088	0.3088	0.3651
0.660	726.085	709.099	693.336	683.743	687.829	691.333	1531.722	0.0681	0.3087	0.3087	0.3649
0.593	710.427	694.585	679.886	671.014	674.810	678.068	1475.621	0.0651	0.3085	0.3085	0.3648
0.525	695.777	681.049	667.389	659.206	662.722	665.742	1415.441	0.0615	0.3084	0.3084	0.3646
0.457	680.538	667.644	655.692	648.579	651.645	654.281	1356.981	0.0528	0.3082	0.3082	0.3644
0.389	666.438	655.536	645.434	639.456	642.041	644.265	1460.906	0.0397	0.3077	0.3080	0.3642
0.330	640.996	638.382	635.962	634.536	635.154	635.685	1259.986	0.0001	0.3020	0.3079	0.3641
0.279	638.382	636.279	634.333	633.187	633.684	634.111	1182.018	0.0001	0.3014	0.3079	0.3641
0.191	634.467	633.395	632.404	631.820	632.073	632.291	976.207	0.0001	0.3004	0.3079	0.3641
0.064	632.168	631.534	630.947	630.602	630.752	630.881	859.823	0.0001	0.2999	0.3079	0.3641

AXIAL FUEL NODE	FUEL-CLAD GAP CONDUCTANCE W/(M**2-K)	C/U HGAP RATIO	LENGTH (M)	ELEV (M)	MASS (KG)	POWER (W)	LINEAR POWER (KW/M)	LINEAR POWER (KW/FT)	BURNUP (A %)	BURNUP (MWD/MT)	FLUENCE (N/CM2)
24	3.35731D+02	0.99571	0.08467	1.6849	2.2764D-02	4.7408D+01	5.5994D-01	1.7067D-01	0.143	1.3714D+03	2.3729D+19
23	3.53776D+02	0.99599	0.08467	1.6002	2.2764D-02	7.5683D+01	8.9389D-01	2.7246D-01	0.229	2.1893D+03	3.7866D+19
22	3.82014D+02	0.99614	0.08467	1.5155	2.2764D-02	1.2141D+02	1.4340D+00	4.3708D-01	0.367	3.5122D+03	6.0709D+19
21	4.64418D+02	0.99722	0.05080	1.4478	1.3659D-02	1.5608D+02	3.0725D+00	9.3649D-01	0.786	7.5252D+03	1.2986D+20
20	5.16480D+02	0.99774	0.05080	1.3970	1.3659D-02	2.0731D+02	4.0810D+00	1.2439D+00	1.045	9.9953D+03	1.7233D+20
19	2.23063D+03	1.00274	0.06773	1.3377	1.8211D-02	1.3169D+03	1.9442D+01	5.9258D+00	4.976	4.7617D+04	8.1574D+20
18	4.76914D+03	1.01598	0.06773	1.2700	1.8211D-02	1.6113D+03	2.3789D+01	7.2510D+00	6.089	5.8265D+04	9.9844D+20
17	6.03762D+03	0.99934	0.06773	1.2023	1.8211D-02	1.8814D+03	2.7777D+01	8.4664D+00	7.109	6.8032D+04	1.1660D+21
16	6.61786D+03	0.99955	0.06773	1.1345	1.8211D-02	2.0508D+03	3.0277D+01	9.2284D+00	7.749	7.4155D+04	1.2708D+21
15	7.35372D+03	1.00106	0.06773	1.0668	1.8211D-02	2.2194D+03	3.2767D+01	9.9875D+00	8.387	8.0254D+04	1.3751D+21
14	8.16626D+03	1.00312	0.06773	0.9991	1.8211D-02	2.3477D+03	3.4661D+01	1.0565D+01	8.871	8.4893D+04	1.4545D+21
13	8.61735D+03	1.00467	0.06773	0.9313	1.8211D-02	2.4014D+03	3.5454D+01	1.0806D+01	9.074	8.6834D+04	1.4878D+21
12	8.82084D+03	1.00571	0.06773	0.8636	1.8211D-02	2.4277D+03	3.5842D+01	1.0925D+01	9.174	8.7785D+04	1.5043D+21
11	8.79815D+03	1.00614	0.06773	0.7959	1.8211D-02	2.4110D+03	3.5596D+01	1.0850D+01	9.111	8.7183D+04	1.4943D+21
10	8.54499D+03	1.00635	0.06773	0.7281	1.8211D-02	2.3650D+03	3.4917D+01	1.0643D+01	8.937	8.5520D+04	1.4662D+21
9	7.83035D+03	1.00534	0.06773	0.6604	1.8211D-02	2.2469D+03	3.3173D+01	1.0111D+01	8.490	8.1248D+04	1.3934D+21
8	7.11295D+03	1.00369	0.06773	0.5927	1.8211D-02	2.0893D+03	3.0847D+01	9.4020D+00	7.895	7.5550D+04	1.2961D+21
7	6.55780D+03	1.00358	0.06773	0.5249	1.8211D-02	1.9365D+03	2.8590D+01	8.7142D+00	7.317	7.0023D+04	1.2016D+21
6	5.27962D+03	1.01079	0.06773	0.4572	1.8211D-02	1.6900D+03	2.4950D+01	7.6049D+00	6.386	6.1109D+04	1.0487D+21
5	2.74499D+03	1.00916	0.06773	0.3895	1.8211D-02	1.4255D+03	2.1046D+01	6.4149D+00	5.387	5.1547D+04	8.8471D+20
4	4.88023D+02	0.99786	0.05080	0.3302	1.3659D-02	2.5555D+02	5.0305D+00	1.5333D+00	1.288	1.2321D+04	2.1269D+20
3	4.37721D+02	0.99747	0.05080	0.2794	1.3659D-02	2.0549D+02	4.0451D+00	1.2330D+00	1.035	9.9074D+03	1.7117D+20
2	3.40364D+02	0.99648	0.12700	0.1905	3.4146D-02	2.6167D+02	2.0604D+00	6.2800D-01	0.527	5.0463D+03	8.7335D+19
1	2.96597D+02	0.99588	0.12700	0.0635	3.4146D-02	1.5470D+02	1.2181D+00	3.7128D-01	0.312	2.9835D+03	5.1677D+19

Figure 8.6.3: SAS4A Fuel-pin Condition Summary Output

CHANNEL 1 SAS4A 3.0 DEFORM-4 TEST PROBLEM FOR SAS4A/SASSYS-1 VERSION 3.0 01/ 9/94 18:20:20 PAGE 119
JOB: 21122/pepper ARGONNE NATIONAL LABORATORY, JANUARY, 1994 USER: 22157/cahalan

*** STEADY STATE TIME (SEC,HRS) = 5.04576D+07 1.40160D+04 ***

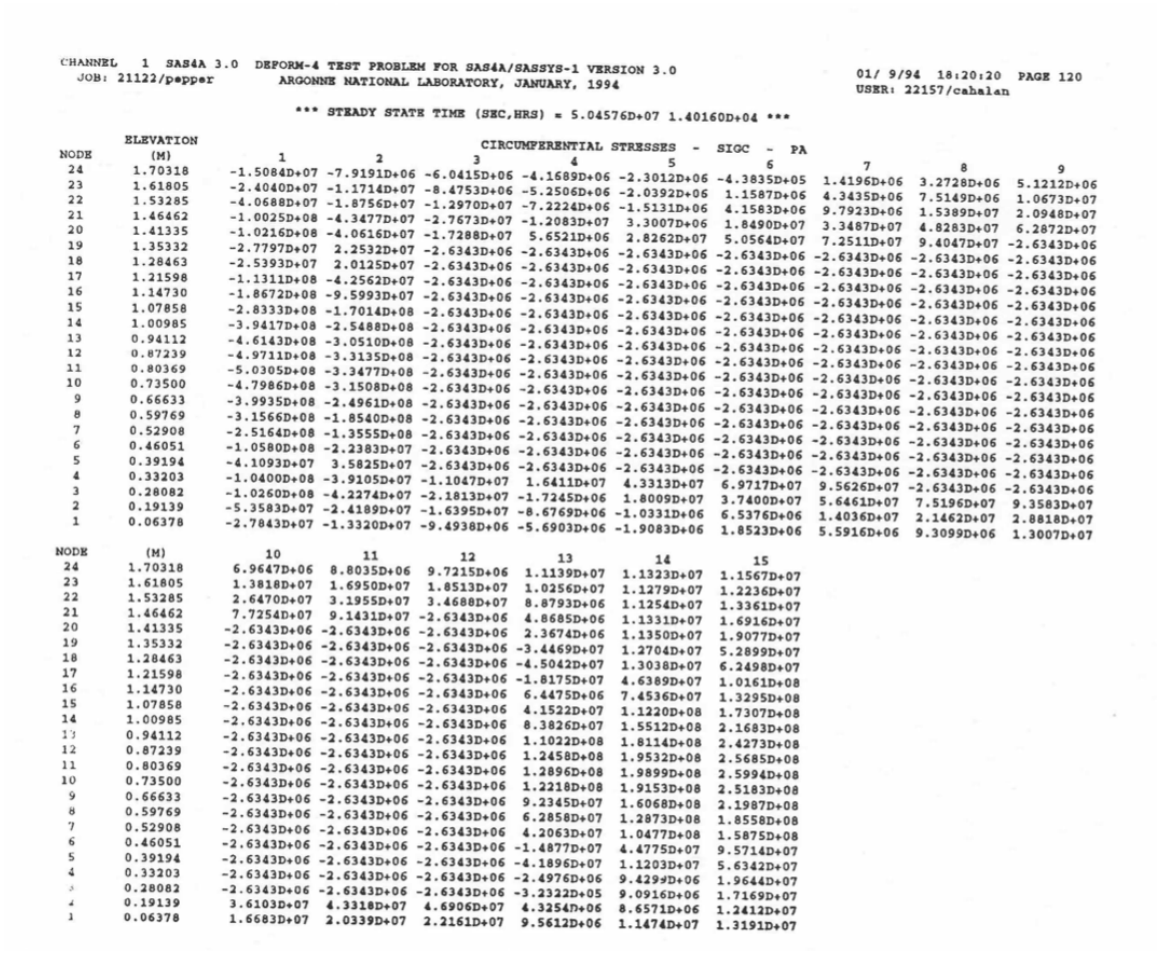
RADIAL STRESSES - SIGR - PA

NODE	ELEVATION (M)	1	2	3	4	5	6	7	8	9
24	1.70318	-2.6343D+06	-8.5459D+06	-7.9197D+06	-7.2943D+06	-6.6698D+06	-6.0463D+06	-5.4237D+06	-4.8022D+06	-4.1816D+06
23	1.61805	-2.6343D+06	-1.2796D+07	-1.1716D+07	-1.0637D+07	-9.5615D+06	-8.4884D+06	-7.4181D+06	-6.3504D+06	-5.2853D+06
22	1.53285	-2.6343D+06	-2.0693D+07	-1.8761D+07	-1.6836D+07	-1.4918D+07	-1.3007D+07	-1.1104D+07	-9.2089D+06	-7.3211D+06
21	1.46462	-2.6343D+06	-4.8788D+07	-4.3503D+07	-3.8254D+07	-3.3045D+07	-2.7877D+07	-2.2748D+07	-1.7659D+07	-1.2610D+07
20	1.41335	-2.6343D+06	-4.8471D+07	-4.0664D+07	-3.2920D+07	-2.5248D+07	-1.7642D+07	-1.0103D+07	-2.6343D+06	-2.6343D+06
19	1.35332	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06
18	1.28463	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06
17	1.21598	-2.6343D+06	-2.2050D+07	-1.6685D+07	-1.4162D+07	-1.2638D+07	-1.1583D+07	-1.0770D+07	-1.0122D+07	-9.6064D+06
16	1.14730	-2.6343D+06	-3.8150D+07	-2.8429D+07	-2.3860D+07	-2.1081D+07	-1.9159D+07	-1.7672D+07	-1.6470D+07	-1.5513D+07
15	1.07858	-2.6343D+06	-5.9458D+07	-4.4046D+07	-3.6786D+07	-3.2353D+07	-2.9285D+07	-2.6929D+07	-2.4991D+07	-2.3440D+07
14	1.00985	-2.6343D+06	-8.3686D+07	-6.1834D+07	-5.1515D+07	-4.5203D+07	-4.0830D+07	-3.7497D+07	-3.4727D+07	-3.2497D+07
13	0.94112	-2.6343D+06	-9.8688D+07	-7.2745D+07	-6.0519D+07	-5.3043D+07	-4.7865D+07	-4.3917D+07	-4.0633D+07	-3.7994D+07
12	0.87239	-2.6343D+06	-1.0721D+08	-7.8847D+07	-6.5521D+07	-5.7383D+07	-5.1752D+07	-4.7440D+07	-4.3866D+07	-4.1000D+07
11	0.80369	-2.6343D+06	-1.1038D+08	-8.0825D+07	-6.7055D+07	-5.8674D+07	-5.2884D+07	-4.8399D+07	-4.4729D+07	-4.1803D+07
10	0.73500	-2.6343D+06	-1.0727D+08	-7.8189D+07	-6.4768D+07	-5.6631D+07	-5.1018D+07	-4.6609D+07	-4.3070D+07	-4.0263D+07
9	0.66633	-2.6343D+06	-9.1078D+07	-6.5919D+07	-5.4491D+07	-4.7611D+07	-4.2818D+07	-3.9058D+07	-3.6125D+07	-3.3806D+07
8	0.59769	-2.6343D+06	-7.3833D+07	-5.3028D+07	-4.3738D+07	-3.8181D+07	-3.4252D+07	-3.1243D+07	-2.8944D+07	-2.7130D+07
7	0.52908	-2.6343D+06	-6.1019D+07	-4.3282D+07	-3.5616D+07	-3.1035D+07	-2.7814D+07	-2.5411D+07	-2.3588D+07	-2.2150D+07
6	0.46051	-2.6343D+06	-2.4708D+07	-1.7434D+07	-1.4505D+07	-1.2795D+07	-1.1621D+07	-1.0764D+07	-1.0115D+07	-9.6029D+06
5	0.39194	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06
4	0.33203	-2.6343D+06	-4.8579D+07	-3.9178D+07	-2.9878D+07	-2.0689D+07	-1.1610D+07	-2.6343D+06	-2.6343D+06	-2.6343D+06
3	0.28082	-2.6343D+06	-4.9168D+07	-4.2318D+07	-3.5530D+07	-2.8813D+07	-2.2166D+07	-1.5587D+07	-9.0770D+06	-2.6343D+06
2	0.19139	-2.6343D+06	-2.6802D+07	-2.4198D+07	-2.1607D+07	-1.9030D+07	-1.6467D+07	-1.3920D+07	-1.1387D+07	-8.8681D+06
1	0.06378	-2.6343D+06	-1.4599D+07	-1.3322D+07	-1.2049D+07	-1.0780D+07	-9.5149D+06	-8.2543D+06	-6.9980D+06	-5.7460D+06

NODE	(M)	10	11	12	13	14	15	GAP	PFCI	PEXT
24	1.70318	-3.5619D+06	-2.9433D+06	-2.6343D+06	-2.6343D+06	-1.4767D+06	-4.8080D+05	9.0104D-05	-2.6343D+06	3.6419D+05
23	1.61805	-4.2229D+06	-3.1632D+06	-2.6343D+06	-2.6343D+06	-1.5153D+06	-4.9241D+05	8.8267D-05	-2.6343D+06	3.7865D+05
22	1.53285	-5.4408D+06	-3.5680D+06	-2.6343D+06	-2.6343D+06	-1.5739D+06	-5.0402D+05	8.5360D-05	-2.6343D+06	3.9310D+05
21	1.46462	-7.6018D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06	-1.7381D+06	-5.1563D+05	7.6460D-05	-2.6343D+06	4.0467D+05
20	1.41335	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06	-1.8417D+06	-5.2724D+05	7.0105D-05	-2.6343D+06	4.1336D+05
19	1.35332	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06	-3.3231D+06	-5.3889D+05	7.5047D-06	-2.6343D+06	4.2354D+05
18	1.28463	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06	-3.7505D+06	-5.5061D+05	4.9918D-07	-2.6343D+06	4.3521D+05
17	1.21598	-9.1859D+06	-8.8355D+06	-8.6812D+06	-8.6812D+06	-6.7844D+06	-5.6238D+05	0.0000D+00	-8.6812D+06	4.4695D+05
16	1.14730	-1.4733D+07	-1.4084D+07	-1.3799D+07	-1.3799D+07	-9.2769D+06	-5.7422D+05	0.0000D+00	-1.3799D+07	4.5875D+05
15	1.07858	-2.2174D+07	-2.1122D+07	-2.0660D+07	-2.0660D+07	-1.2534D+07	-5.8606D+05	0.0000D+00	-2.0660D+07	4.7057D+05
14	1.00985	-3.0677D+07	-2.9165D+07	-2.8502D+07	-2.8502D+07	-1.6164D+07	-5.9645D+05	0.0000D+00	-2.8502D+07	4.8238D+05
13	0.94112	-3.5839D+07	-3.4050D+07	-3.3265D+07	-3.3265D+07	-1.8341D+07	-6.0538D+05	0.0000D+00	-3.3265D+07	4.9418D+05
12	0.87239	-3.8663D+07	-3.6723D+07	-3.5872D+07	-3.5872D+07	-1.9539D+07	-6.2079D+05	0.0000D+00	-3.5872D+07	5.0597D+05
11	0.80369	-3.9419D+07	-3.7441D+07	-3.6573D+07	-3.6573D+07	-1.9845D+07	-6.4270D+05	0.0000D+00	-3.6573D+07	5.1776D+05
10	0.73500	-3.7978D+07	-3.6081D+07	-3.5248D+07	-3.5248D+07	-1.9224D+07	-6.6771D+05	0.0000D+00	-3.5248D+07	5.2954D+05
9	0.66633	-3.1920D+07	-3.0353D+07	-2.9664D+07	-2.9664D+07	-1.6635D+07	-6.9582D+05	0.0000D+00	-2.9664D+07	5.4130D+05
8	0.59769	-2.5655D+07	-2.4428D+07	-2.3887D+07	-2.3887D+07	-1.3902D+07	-7.2390D+05	0.0000D+00	-2.3887D+07	5.5305D+05
7	0.52908	-2.0979D+07	-2.0004D+07	-1.9574D+07	-1.9574D+07	-1.1814D+07	-7.5197D+05	0.0000D+00	-1.9574D+07	5.6478D+05
6	0.46051	-9.1854D+06	-8.8368D+06	-8.6830D+06	-8.6830D+06	-6.7106D+06	-7.8001D+05	0.0000D+00	-8.6830D+06	5.7648D+05
5	0.39194	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06	-3.6958D+06	-8.0801D+05	3.1913D-06	-2.6343D+06	5.8815D+05
4	0.33203	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.1250D+06	-8.4085D+05	5.9128D-05	-2.6343D+06	5.9832D+05
3	0.28082	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.0483D+06	-8.7852D+05	6.4626D-05	-2.6343D+06	6.0701D+05
2	0.19139	-6.3639D+06	-3.8741D+06	-2.6343D+06	-2.6343D+06	-1.8724D+06	-9.1620D+05	7.4914D-05	-2.6343D+06	6.2219D+05
1	0.06378	-4.4982D+06	-3.2546D+06	-2.6343D+06	-2.6343D+06	-1.5362D+06	-4.6731D+05	7.9379D-05	-2.6343D+06	6.4388D+05

Figure 8.6.4: DEFORM-4 Radial Stress Map

The radial stress at each fuel and cladding node. Also printed is the fuel-cladding gap, GAP, the fuel-cladding interface pressure, PFCI, and the coolant pressure, PEXT. All stresses and pressures are in Pascal and the gap is in meters.



CHANNEL 1 SAS4A 3.0 DEFORM-4 TEST PROBLEM FOR SAS4A/SASSYS-1 VERSION 3.0 01/ 9/94 18:20:20 PAGE 120
JOB: 21122/pepper ARGONNE NATIONAL LABORATORY, JANUARY, 1994 USER: 22157/cahalan

*** STEADY STATE TIME (SEC,HRS) = 5.04576D+07 1.40160D+04 ***

NODE	ELEVATION (M)	CIRCUMFERENTIAL STRESSES - SIGC - PA 1	2	3	4	5	6	7	8	9
24	1.70318	-1.5084D+07	-7.9191D+06	-6.0415D+06	-4.1689D+06	-2.3012D+06	-4.3835D+05	1.4196D+06	3.2728D+06	5.1212D+06
23	1.61805	-2.4040D+07	-1.1714D+07	-8.4753D+06	-5.2506D+06	-2.0392D+06	1.1587D+06	4.3435D+06	7.5149D+06	1.0673D+07
22	1.53285	-4.0688D+07	-1.8756D+07	-1.2970D+07	-7.2224D+06	-1.5131D+06	4.1583D+06	9.7923D+06	1.5389D+07	2.0948D+07
21	1.46462	-1.0025D+08	-4.3477D+07	-2.7673D+07	-1.2083D+07	3.3007D+06	1.8490D+07	3.3487D+07	4.8283D+07	6.2872D+07
20	1.41335	-1.0216D+08	-4.0616D+07	-1.7288D+07	5.6521D+06	2.8262D+07	5.0564D+07	7.2511D+07	9.4047D+07	-2.6343D+06
19	1.35332	-2.7797D+07	2.2532D+07	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06
18	1.28463	-2.5393D+07	2.0125D+07	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06
17	1.21598	-1.1311D+08	-4.2562D+07	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06
16	1.14730	-1.8672D+08	-9.5993D+07	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06
15	1.07858	-2.8333D+08	-1.7014D+08	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06
14	1.00985	-3.9417D+08	-2.5488D+08	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06
13	0.94112	-4.6143D+08	-3.0510D+08	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06
12	0.87239	-4.9711D+08	-3.3135D+08	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06
11	0.80369	-5.0305D+08	-3.3477D+08	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06
10	0.73500	-4.7986D+08	-3.1508D+08	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06
9	0.66633	-3.9935D+08	-2.4961D+08	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06
8	0.59769	-3.1566D+08	-1.8540D+08	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06
7	0.52908	-2.5164D+08	-1.3555D+08	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06
6	0.46051	-1.0580D+08	-2.2383D+07	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06
5	0.39194	-4.1093D+07	3.5825D+07	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06
4	0.33203	-1.0400D+08	-3.9105D+07	-1.1047D+07	1.6411D+07	4.3313D+07	6.9717D+07	9.5626D+07	-2.6343D+06	-2.6343D+06
3	0.28082	-1.0260D+08	-4.2274D+07	-2.1813D+07	-1.7245D+06	1.8009D+07	3.7400D+07	5.6461D+07	7.5196D+07	9.3583D+07
2	0.19139	-5.3583D+07	-2.4189D+07	-1.6395D+07	-8.6769D+06	-1.0331D+06	6.5376D+06	1.4036D+07	2.1462D+07	2.8818D+07
1	0.06378	-2.7843D+07	-1.3320D+07	-9.4938D+06	-5.6903D+06	-1.9083D+06	1.8523D+06	5.5916D+06	9.3099D+06	1.3007D+07

NODE	(M)	10	11	12	13	14	15
24	1.70318	6.9647D+06	8.8035D+06	9.7215D+06	1.1139D+07	1.1323D+07	1.1567D+07
23	1.61805	1.3818D+07	1.6950D+07	1.8513D+07	1.0256D+07	1.1279D+07	1.2236D+07
22	1.53285	2.6470D+07	3.1955D+07	3.4688D+07	8.8793D+06	1.1254D+07	1.3361D+07
21	1.46462	7.7254D+07	9.1431D+07	-2.6343D+06	4.8685D+06	1.1331D+07	1.6916D+07
20	1.41335	-2.6343D+06	-2.6343D+06	-2.6343D+06	2.3674D+06	1.1350D+07	1.9077D+07
19	1.35332	-2.6343D+06	-2.6343D+06	-2.6343D+06	-3.4469D+07	1.2704D+07	5.2899D+07
18	1.28463	-2.6343D+06	-2.6343D+06	-2.6343D+06	-4.5042D+07	1.3038D+07	6.2498D+07
17	1.21598	-2.6343D+06	-2.6343D+06	-2.6343D+06	-1.8175D+07	4.6389D+07	1.0161D+08
16	1.14730	-2.6343D+06	-2.6343D+06	-2.6343D+06	6.4475D+06	7.4536D+07	1.3295D+08
15	1.07858	-2.6343D+06	-2.6343D+06	-2.6343D+06	4.1522D+07	1.1220D+08	1.7307D+08
14	1.00985	-2.6343D+06	-2.6343D+06	-2.6343D+06	8.3826D+07	1.5512D+08	2.1683D+08
13	0.94112	-2.6343D+06	-2.6343D+06	-2.6343D+06	1.1022D+08	1.8114D+08	2.4273D+08
12	0.87239	-2.6343D+06	-2.6343D+06	-2.6343D+06	1.2458D+08	1.9532D+08	2.5685D+08
11	0.80369	-2.6343D+06	-2.6343D+06	-2.6343D+06	1.2896D+08	1.9899D+08	2.5994D+08
10	0.73500	-2.6343D+06	-2.6343D+06	-2.6343D+06	1.2218D+08	1.9153D+08	2.5183D+08
9	0.66633	-2.6343D+06	-2.6343D+06	-2.6343D+06	9.2345D+07	1.6068D+08	2.1987D+08
8	0.59769	-2.6343D+06	-2.6343D+06	-2.6343D+06	6.2858D+07	1.2873D+08	1.8558D+08
7	0.52908	-2.6343D+06	-2.6343D+06	-2.6343D+06	4.2063D+07	1.0477D+08	1.5875D+08
6	0.46051	-2.6343D+06	-2.6343D+06	-2.6343D+06	-1.4877D+07	4.4775D+07	9.5714D+07
5	0.39194	-2.6343D+06	-2.6343D+06	-2.6343D+06	-4.1896D+07	1.1203D+07	5.6342D+07
4	0.33203	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.4976D+06	9.4299D+06	1.9644D+07
3	0.28082	-2.6343D+06	-2.6343D+06	-2.6343D+06	-3.2322D+05	9.0916D+06	1.7169D+07
2	0.19139	3.6103D+07	4.3318D+07	4.6906D+07	4.3254D+06	8.6571D+06	1.2412D+07
1	0.06378	1.6683D+07	2.0339D+07	2.2161D+07	9.5612D+06	1.1474D+07	1.3191D+07

Figure 8.6.5: DEFORM-4 Circumferential Stress Map

The circumferential stress at each fuel and clad node boundary, in pascals.

The axial stress at each fuel and clad node boundary in pascals.

The fission-gas retention fraction in each fuel cell. The column titled 'FG RET (KG)' is the retained fission-gas total in kg, 'FG PRD (KG)' is the total kg of fission gas produced, 'M-FG/M-FUEL' is the kg of fission gas retained divided by the mass of fuel, 'FRAC RELS' is the fractional release, and 'NUET/M2' is the fluence, for each axial segment.

The total porosity fraction in the fuel cells, i.e., retained fission-gas porosity plus remaining as-fabricated porosity.

The fission-gas-induced porosity in each fuel cell.

The grain size in each fuel cell, in meters.

The crack volume fraction in each fuel cell.

Mesh points in the fuel and the inner and outer clad dimensions. Also shown are the boundaries between the unrestructured and equiaxed region, IUNEQA, and between the equiaxed and columnar regions, IEQCOL.

CHANNEL 1 SAS4A 3.0 DEFORM-4 TEST PROBLEM FOR SAS4A/SASSYS-1 VERSION 3.0 01/ 9/94 18:20:20 PAGE 121
JOB: 21122/pepper ARGONNE NATIONAL LABORATORY, JANUARY, 1994 USER: 22157/cahalan

*** STEADY STATE TIME (SEC,HRS) = 5.04576D+07 1.40160D+04 ***

AXIAL STRESSES - SIGZ - PA

NODE	ELEVATION (M)	1	2	3	4	5	6	7	8	9
24	1.70318	-1.5719D+07	-1.4466D+07	-1.1962D+07	-9.4637D+06	-6.9715D+06	-4.4852D+06	-2.0046D+06	4.7009D+05	2.9390D+06
23	1.61805	-2.4647D+07	-2.2483D+07	-1.8163D+07	-1.3860D+07	-9.5734D+06	-5.3024D+06	-1.0473D+06	3.1919D+06	7.4152D+06
22	1.53285	-4.1243D+07	-3.7370D+07	-2.9651D+07	-2.1978D+07	-1.4351D+07	-6.7693D+06	7.6752D+05	8.2594D+06	1.5706D+07
21	1.46462	-1.0392D+08	-9.3300D+07	-7.2210D+07	-5.1372D+07	-3.0779D+07	-1.0421D+07	9.7046D+06	2.9590D+07	4.9227D+07
20	1.41335	-1.3575D+08	-1.2004D+08	-8.8907D+07	-5.8224D+07	-2.7942D+07	1.9657D+06	3.1452D+07	6.0457D+07	-2.6343D+06
19	1.35332	-4.2610D+08	-3.7577D+08	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06
18	1.28463	-4.0933D+08	-3.6381D+08	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06
17	1.21598	-5.1971D+08	-4.6858D+08	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06
16	1.14730	-5.8643D+08	-5.3123D+08	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06
15	1.07858	-6.2917D+08	-5.7280D+08	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06
14	1.00985	-6.7108D+08	-6.1284D+08	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06
13	0.94112	-6.9110D+08	-6.3083D+08	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06
12	0.87239	-7.0356D+08	-6.4237D+08	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06
11	0.80369	-6.9983D+08	-6.3929D+08	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06
10	0.73500	-6.8369D+08	-6.2355D+08	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06
9	0.66633	-6.4136D+08	-5.8006D+08	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06
8	0.59769	-5.6113D+08	-5.0207D+08	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06
7	0.52908	-4.6890D+08	-4.1119D+08	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06
6	0.46051	-3.6387D+08	-3.0252D+08	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06
5	0.39194	-4.7179D+08	-3.9488D+08	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06	-2.6343D+06
4	0.33203	-2.5258D+08	-2.3363D+08	-1.9617D+08	-1.5941D+08	-1.2332D+08	-8.7839D+07	-5.2954D+07	-2.6343D+06	-2.6343D+06
3	0.28082	-2.1416D+08	-2.0037D+08	-1.7306D+08	-1.4618D+08	-1.1973D+08	-9.3692D+07	-6.8052D+07	-4.2807D+07	-1.7977D+07
2	0.19139	-1.1661D+08	-1.1139D+08	-1.0099D+08	-9.0681D+07	-8.0460D+07	-7.0327D+07	-6.0281D+07	-5.0322D+07	-4.0448D+07
1	0.06378	-7.1900D+07	-6.9341D+07	-6.4239D+07	-5.9162D+07	-5.4111D+07	-4.9085D+07	-4.4085D+07	-3.9111D+07	-3.4162D+07

NODE	(M)	10	11	12	13	14	15
24	1.70318	5.4022D+06	7.8596D+06	9.0866D+06	-2.0518D+06	-7.1062D+05	5.3002D+05
23	1.61805	1.1623D+07	1.5815D+07	1.7906D+07	-2.8771D+06	-7.3602D+05	1.2446D+06
22	1.53285	2.3109D+07	3.0467D+07	3.4133D+07	-4.1971D+06	-7.6211D+05	2.4153D+06
21	1.46462	6.8618D+07	8.7762D+07	-2.6343D+06	-8.1507D+06	-7.9178D+05	6.0152D+06
20	1.41335	-2.6343D+06	-2.6343D+06	-2.6343D+06	-1.0595D+07	-8.1935D+05	8.2224D+06
19	1.35332	-2.6343D+06	-2.6343D+06	-2.6343D+06	-4.7374D+07	-8.8936D+05	4.2090D+07
18	1.28463	-2.6343D+06	-2.6343D+06	-2.6343D+06	-5.7889D+07	-9.2476D+05	5.1734D+07
17	1.21598	-2.6343D+06	-2.6343D+06	-2.6343D+06	-6.7422D+07	-9.6129D+05	6.0479D+07
16	1.14730	-2.6343D+06	-2.6343D+06	-2.6343D+06	-7.3601D+07	-9.9019D+05	6.6124D+07
15	1.07858	-2.6343D+06	-2.6343D+06	-2.6343D+06	-7.9820D+07	-1.0175D+06	7.1802D+07
14	1.00985	-2.6343D+06	-2.6343D+06	-2.6343D+06	-8.4678D+07	-1.0418D+06	7.6235D+07
13	0.94112	-2.6343D+06	-2.6343D+06	-2.6343D+06	-8.6910D+07	-1.0620D+06	7.8257D+07
12	0.87239	-2.6343D+06	-2.6343D+06	-2.6343D+06	-8.8169D+07	-1.0946D+06	7.9355D+07
11	0.80369	-2.6343D+06	-2.6343D+06	-2.6343D+06	-8.7900D+07	-1.1350D+06	7.9010D+07
10	0.73500	-2.6343D+06	-2.6343D+06	-2.6343D+06	-8.6563D+07	-1.1809D+06	7.7668D+07
9	0.66633	-2.6343D+06	-2.6343D+06	-2.6343D+06	-8.2598D+07	-1.2320D+06	7.3894D+07
8	0.59769	-2.6343D+06	-2.6343D+06	-2.6343D+06	-7.7137D+07	-1.2834D+06	6.8745D+07
7	0.52908	-2.6343D+06	-2.6343D+06	-2.6343D+06	-7.1800D+07	-1.3338D+06	6.3707D+07
6	0.46051	-2.6343D+06	-2.6343D+06	-2.6343D+06	-6.3007D+07	-1.3821D+06	5.5487D+07
5	0.39194	-2.6343D+06	-2.6343D+06	-2.6343D+06	-5.3472D+07	-1.4340D+06	4.6592D+07
4	0.33203	-2.6343D+06	-2.6343D+06	-2.6343D+06	-1.3912D+07	-1.4752D+06	1.0023D+07
3	0.28082	-2.6343D+06	-2.6343D+06	-2.6343D+06	-1.1552D+07	-1.5517D+06	7.6951D+06
2	0.19139	-3.0658D+07	-2.0954D+07	-1.6126D+07	-6.7186D+06	-1.6250D+06	3.0857D+06
1	0.06378	-2.9237D+07	-2.4338D+07	-2.1896D+07	-3.6884D+06	-6.7705D+05	2.1082D+06

Figure 8.6.6: DEFORM-4 Axial Stress Map

CHANNEL 1 SAS4A 3.0 DEFORM-4 TEST PROBLEM FOR SAS4A/SASSYS-1 VERSION 3.0 01/ 9/94 18:20:20 PAGE 122
JOB: 21122/pepper ARGONNE NATIONAL LABORATORY, JANUARY, 1994 USER: 22157/cahalan

*** STEADY STATE TIME (SEC,HRS) = 5.04576D+07 1.40160D+04 ***

	ELEVATION	FISSION GAS RETENTION								
NODE	(M)	1	2	3	4	5	6	7	8	9
24	1.70318	9.7328D-01	9.7381D-01	9.7433D-01	9.7485D-01	9.7535D-01	9.7585D-01	9.7633D-01	9.7681D-01	9.7728D-01
23	1.61805	9.5082D-01	9.5227D-01	9.5367D-01	9.5505D-01	9.5638D-01	9.5769D-01	9.5895D-01	9.6019D-01	9.6139D-01
22	1.53285	8.9132D-01	8.9581D-01	9.0015D-01	9.0434D-01	9.0839D-01	9.1229D-01	9.1606D-01	9.1969D-01	9.2319D-01
21	1.46462	5.4557D-01	5.6917D-01	5.9245D-01	6.1533D-01	6.3773D-01	6.5957D-01	6.8078D-01	7.0128D-01	7.2104D-01
20	1.41335	3.2748D-01	3.5578D-01	3.8529D-01	4.1579D-01	4.4709D-01	4.7890D-01	5.1089D-01	5.4276D-01	5.7421D-01
19	1.35332	2.4669D-02	2.7565D-02	3.2613D-02	3.9618D-02	4.9635D-02	6.5016D-02	9.0242D-02	1.3223D-01	2.0066D-01
18	1.28463	3.1813D-02	3.5839D-02	4.3628D-02	5.5289D-02	7.2578D-02	1.0066D-01	1.5011D-01	2.3517D-01	3.6590D-01
17	1.21598	2.4433D-02	2.7798D-02	3.4654D-02	4.5654D-02	6.3093D-02	9.4473D-02	1.5952D-01	2.8584D-01	4.7457D-01
16	1.14730	2.0185D-02	2.2976D-02	2.8652D-02	3.7760D-02	5.2253D-02	7.8645D-02	1.3850D-01	2.7287D-01	4.8982D-01
15	1.07858	1.7509D-02	1.9952D-02	2.4950D-02	3.2987D-02	4.5815D-02	6.8573D-02	1.2148D-01	2.5562D-01	4.9442D-01
14	1.00985	1.6452D-02	1.8800D-02	2.3641D-02	3.1484D-02	4.4111D-02	6.6181D-02	1.1729D-01	2.5404D-01	5.0901D-01
13	0.94112	1.6500D-02	1.8921D-02	2.3931D-02	3.2095D-02	4.5330D-02	6.8639D-02	1.2321D-01	2.7007D-01	5.3605D-01
12	0.87239	1.6768D-02	1.9293D-02	2.4523D-02	3.3086D-02	4.7042D-02	7.2011D-02	1.3162D-01	2.9113D-01	5.6545D-01
11	0.80369	1.7659D-02	2.0409D-02	2.6087D-02	3.5405D-02	5.0652D-02	7.9002D-02	1.4900D-01	3.2858D-01	6.0647D-01
10	0.73500	1.9076D-02	2.2147D-02	2.8456D-02	3.8816D-02	5.5837D-02	8.9259D-02	1.7420D-01	3.7469D-01	6.4810D-01
9	0.66633	2.1930D-02	2.5609D-02	3.3077D-02	4.5340D-02	6.6122D-02	1.1057D-01	2.2184D-01	4.4072D-01	6.9310D-01
8	0.59769	2.7410D-02	3.2251D-02	4.1986D-02	5.8137D-02	8.7404D-02	1.5257D-01	2.9421D-01	5.1344D-01	7.3182D-01
7	0.52908	3.5481D-02	4.2234D-02	5.5607D-02	7.8051D-02	1.1966D-01	2.0401D-01	3.5396D-01	5.5283D-01	7.4206D-01
6	0.46051	4.7317D-02	5.7169D-02	7.4948D-02	1.0277D-01	1.4893D-01	2.2675D-01	3.4677D-01	5.0530D-01	6.7313D-01
5	0.39194	3.5822D-02	4.4034D-02	5.6730D-02	7.6040D-02	1.0743D-01	1.5907D-01	2.4070D-01	3.5943D-01	5.0775D-01
4	0.33203	3.8697D-01	4.2504D-01	4.6404D-01	5.0348D-01	5.4287D-01	5.8177D-01	6.1967D-01	6.5609D-01	6.9075D-01
3	0.28082	5.9890D-01	6.2884D-01	6.5783D-01	6.8569D-01	7.1231D-01	7.3759D-01	7.6146D-01	7.8387D-01	8.0479D-01
2	0.19139	9.4218D-01	9.4593D-01	9.4947D-01	9.5280D-01	9.5595D-01	9.5890D-01	9.6169D-01	9.6430D-01	9.6676D-01
1	0.06378	9.8828D-01	9.8882D-01	9.8934D-01	9.8984D-01	9.9032D-01	9.9077D-01	9.9121D-01	9.9163D-01	9.9203D-01

NODE	(M)	10	11	FG RET(KG)	FG PRD(KG)	M-FG/M-FUEL	FRAC RELS	NEUT/M**2
24	1.70318	9.7774D-01	9.7819D-01	3.9269D-06	4.0243D-06	1.7250D-04	2.4192D-02	2.3729D+23
23	1.61805	9.6256D-01	9.6370D-01	6.1516D-06	6.4243D-06	2.7023D-04	4.2459D-02	3.7866D+23
22	1.53285	9.2656D-01	9.2980D-01	9.3961D-06	1.0306D-05	4.1275D-04	8.8296D-02	6.0709D+23
21	1.46462	7.4000D-01	7.5815D-01	8.7035D-06	1.3249D-05	6.3722D-04	3.4308D-01	1.2986D+24
20	1.41335	6.0504D-01	6.3506D-01	8.4417D-06	1.7598D-05	6.1805D-04	5.2030D-01	1.7233D+24
19	1.35332	3.0467D-01	4.4168D-01	1.3139D-05	1.1178D-04	7.2147D-04	8.8246D-01	8.1574D+24
18	1.28463	5.3233D-01	6.9802D-01	2.6759D-05	1.3678D-04	1.4694D-03	8.0436D-01	9.9844D+24
17	1.21598	6.7612D-01	8.3116D-01	3.6564D-05	1.5970D-04	2.0078D-03	7.7105D-01	1.1660D+25
16	1.14730	7.1321D-01	8.6714D-01	3.9661D-05	1.7408D-04	2.1778D-03	7.7216D-01	1.2708D+25
15	1.07858	7.3767D-01	8.9101D-01	4.2497D-05	1.8840D-04	2.3335D-03	7.7443D-01	1.3751D+25
14	1.00985	7.6181D-01	9.0933D-01	4.5622D-05	1.9929D-04	2.5051D-03	7.7107D-01	1.4545D+25
13	0.94112	7.8530D-01	9.2225D-01	4.8370D-05	2.0384D-04	2.6560D-03	7.6271D-01	1.4878D+25
12	0.87239	8.0666D-01	9.3244D-01	5.0806D-05	2.0607D-04	2.7898D-03	7.5346D-01	1.5043D+25
11	0.80369	8.3045D-01	9.4182D-01	5.3330D-05	2.0466D-04	2.9284D-03	7.3942D-01	1.4943D+25
10	0.73500	8.5102D-01	9.4884D-01	5.5539D-05	2.0076D-04	3.0497D-03	7.2335D-01	1.4662D+25
9	0.66633	8.6774D-01	9.5246D-01	5.7053D-05	1.9073D-04	3.1328D-03	7.0087D-01	1.3934D+25
8	0.59769	8.7851D-01	9.5280D-01	5.8180D-05	1.7735D-04	3.1947D-03	6.7196D-01	1.2961D+25
7	0.52908	8.7386D-01	9.4605D-01	5.7747D-05	1.6438D-04	3.1709D-03	6.4870D-01	1.2016D+25
6	0.46051	8.1151D-01	9.0329D-01	4.9098D-05	1.4345D-04	2.6960D-03	6.5774D-01	1.0487D+25
5	0.39194	6.5952D-01	7.8689D-01	3.1729D-05	1.2101D-04	1.7422D-03	7.3779D-01	8.8471D+24
4	0.33203	7.2342D-01	7.5393D-01	1.2533D-05	2.1692D-05	9.1757D-04	4.2224D-01	2.1269D+24
3	0.28082	8.2421D-01	8.4216D-01	1.2763D-05	1.7443D-05	9.3446D-04	2.6829D-01	1.7117D+24
2	0.19139	9.6907D-01	9.7123D-01	2.1282D-05	2.2212D-05	6.2326D-04	4.1843D-02	8.7335D+23
1	0.06378	9.9241D-01	9.9278D-01	1.3010D-05	1.3132D-05	3.8099D-04	9.3087D-03	5.1677D+23

Figure 8.6.7: DEFORM-4 Fission-gas Retention Fraction Map

CHANNEL 1 SAS4A 3.0 DEFORM-4 TEST PROBLEM FOR SAS4A/SASSYS-1 VERSION 3.0 01/ 9/94 18:20:20 PAGE 123
JOB: 21122/pepper ARGONNE NATIONAL LABORATORY, JANUARY, 1994 USER: 22157/cahalan

*** STEADY STATE TIME (SEC,HRS) = 5.04576D+07 1.40160D+04 ***

NODE	ELEVATION (M)	TOTAL POROSITY DISTRIBUTION 1	2	3	4	5	6	7	8	9
24	1.70318	1.3202D-01	1.3202D-01	1.3202D-01	1.3202D-01	1.3202D-01	1.3202D-01	1.3202D-01	1.3202D-01	1.3202D-01
23	1.61805	1.3207D-01	1.3207D-01	1.3207D-01	1.3207D-01	1.3207D-01	1.3207D-01	1.3207D-01	1.3207D-01	1.3207D-01
22	1.53285	1.3219D-01	1.3219D-01	1.3219D-01	1.3220D-01	1.3221D-01	1.3221D-01	1.3222D-01	1.3221D-01	1.3221D-01
21	1.46462	1.3261D-01	1.3266D-01	1.3276D-01	1.3289D-01	1.3310D-01	1.3353D-01	1.3359D-01	1.3360D-01	1.3360D-01
20	1.41335	1.3285D-01	1.3299D-01	1.3326D-01	1.3380D-01	1.3534D-01	1.3566D-01	1.3588D-01	1.3530D-01	1.3447D-01
19	1.35332	5.2549D-02	6.6479D-02	5.5996D-02	6.4395D-02	8.6003D-02	1.1896D-01	1.4594D-01	1.5611D-01	1.5662D-01
18	1.28463	2.1670D-02	2.9835D-02	4.9599D-02	5.5559D-02	7.8304D-02	1.2085D-01	1.5392D-01	1.6089D-01	1.5780D-01
17	1.21598	2.0440D-02	2.1067D-02	4.5085D-02	5.0732D-02	6.5068D-02	1.1486D-01	1.5724D-01	1.6272D-01	1.5961D-01
16	1.14730	2.0324D-02	2.0773D-02	3.5146D-02	4.1401D-02	5.0121D-02	1.0178D-01	1.5618D-01	1.6310D-01	1.6149D-01
15	1.07858	2.0259D-02	2.0597D-02	2.9350D-02	3.3833D-02	4.0486D-02	8.0354D-02	1.4998D-01	1.6254D-01	1.6271D-01
14	1.00985	2.0220D-02	2.0498D-02	2.6504D-02	3.0205D-02	3.5561D-02	6.5817D-02	1.4355D-01	1.6141D-01	1.6253D-01
13	0.94112	2.0208D-02	2.0468D-02	2.5675D-02	2.9182D-02	3.4235D-02	6.5203D-02	1.4380D-01	1.6109D-01	1.6193D-01
12	0.87239	2.0204D-02	2.0459D-02	2.5424D-02	2.8898D-02	3.3965D-02	6.9608D-02	1.4637D-01	1.6111D-01	1.6137D-01
11	0.80369	2.0211D-02	2.0477D-02	2.5676D-02	2.9364D-02	3.4802D-02	8.4782D-02	1.5185D-01	1.6110D-01	1.6026D-01
10	0.73500	2.0229D-02	2.0522D-02	2.6502D-02	3.0637D-02	3.7093D-02	1.0490D-01	1.5652D-01	1.6112D-01	1.5892D-01
9	0.66633	2.0278D-02	2.0648D-02	2.9212D-02	3.4691D-02	5.7722D-02	1.3237D-01	1.5992D-01	1.6061D-01	1.5664D-01
8	0.59769	2.0379D-02	2.0916D-02	3.4889D-02	4.4879D-02	9.3050D-02	1.5274D-01	1.6085D-01	1.5858D-01	1.5331D-01
7	0.52908	2.2446D-02	3.2177D-02	4.4542D-02	6.7245D-02	1.2018D-01	1.5947D-01	1.6071D-01	1.5645D-01	1.5096D-01
6	0.46051	5.3857D-02	6.2117D-02	7.3953D-02	9.8206D-02	1.3984D-01	1.6051D-01	1.5989D-01	1.5550D-01	1.5118D-01
5	0.39194	1.1919D-01	9.4656D-02	9.7040D-02	1.2005D-01	1.4692D-01	1.5798D-01	1.5779D-01	1.5466D-01	1.5219D-01
4	0.33203	1.3286D-01	1.3298D-01	1.3321D-01	1.3356D-01	1.3427D-01	1.3579D-01	1.3561D-01	1.3501D-01	1.3484D-01
3	0.28082	1.3270D-01	1.3276D-01	1.3286D-01	1.3299D-01	1.3317D-01	1.3347D-01	1.3394D-01	1.3394D-01	1.3378D-01
2	0.19139	1.3220D-01	1.3220D-01	1.3220D-01	1.3220D-01	1.3220D-01	1.3220D-01	1.3221D-01	1.3221D-01	1.3221D-01
1	0.06378	1.3205D-01	1.3205D-01	1.3204D-01	1.3204D-01	1.3204D-01	1.3204D-01	1.3204D-01	1.3204D-01	1.3204D-01

NODE	(M)	10	11
24	1.70318	1.3202D-01	1.3202D-01
23	1.61805	1.3206D-01	1.3206D-01
22	1.53285	1.3220D-01	1.3220D-01
21	1.46462	1.3358D-01	1.3349D-01
20	1.41335	1.3437D-01	1.3426D-01
19	1.35332	1.5432D-01	1.5236D-01
18	1.28463	1.5459D-01	1.5025D-01
17	1.21598	1.5445D-01	1.4762D-01
16	1.14730	1.5565D-01	1.4759D-01
15	1.07858	1.5658D-01	1.4753D-01
14	1.00985	1.5621D-01	1.4672D-01
13	0.94112	1.5514D-01	1.4565D-01
12	0.87239	1.5403D-01	1.4464D-01
11	0.80369	1.5239D-01	1.4343D-01
10	0.73500	1.5069D-01	1.4232D-01
9	0.66633	1.4851D-01	1.4115D-01
8	0.59769	1.4600D-01	1.3990D-01
7	0.52908	1.4459D-01	1.3938D-01
6	0.46051	1.4606D-01	1.4129D-01
5	0.39194	1.4902D-01	1.4515D-01
4	0.33203	1.3467D-01	1.3450D-01
3	0.28082	1.3352D-01	1.3343D-01
2	0.19139	1.3220D-01	1.3219D-01
1	0.06378	1.3204D-01	1.3204D-01

Figure 8.6.8: DEFORM-4 Total Porosity Fraction Map

CHANNEL 1 SAS4A 3.0 DEFORM-4 TEST PROBLEM FOR SAS4A/SASSYS-1 VERSION 3.0 01/ 9/94 18:20:20 PAGE 124
JOB: 21122/pepper ARGONNE NATIONAL LABORATORY, JANUARY, 1994 USER: 22157/cahalan

*** STEADY STATE TIME (SEC,HRS) = 5.04576D+07 1.40160D+04 ***

	ELEVATION	FISSION GAS INDUCED POROSITY								
NODE	(M)	1	2	3	4	5	6	7	8	9
24	1.70318	2.4347D-05	2.4227D-05	2.4166D-05	2.4103D-05	2.4038D-05	2.3970D-05	2.3894D-05	2.3694D-05	2.3415D-05
23	1.61805	6.6628D-05	6.6572D-05	6.6874D-05	6.7177D-05	6.7481D-05	6.7786D-05	6.7784D-05	6.6609D-05	6.5447D-05
22	1.53285	1.8529D-04	1.8822D-04	1.9386D-04	2.0010D-04	2.0707D-04	2.1495D-04	2.1593D-04	2.1106D-04	2.0618D-04
21	1.46462	6.0741D-04	6.6107D-04	7.5530D-04	8.8888D-04	1.1018D-03	1.5287D-03	1.5927D-03	1.6007D-03	1.5962D-03
20	1.41335	8.4920D-04	9.8778D-04	1.2641D-03	1.7985D-03	3.3396D-03	3.6627D-03	3.8761D-03	3.2958D-03	2.4689D-03
19	1.35332	3.8958D-04	9.1581D-04	2.1654D-02	2.3947D-02	2.5848D-02	2.7576D-02	2.8307D-02	2.7416D-02	2.5179D-02
18	1.28463	5.7163D-04	1.2337D-03	2.9599D-02	3.2159D-02	3.4032D-02	3.4542D-02	3.3570D-02	3.0518D-02	2.5954D-02
17	1.21598	4.3999D-04	1.0674D-03	2.5085D-02	3.0732D-02	3.5379D-02	3.7629D-02	3.6603D-02	3.1807D-02	2.7675D-02
16	1.14730	3.2447D-04	7.7284D-04	1.5146D-02	2.1401D-02	2.8783D-02	3.3531D-02	3.6352D-02	3.2029D-02	2.9536D-02
15	1.07858	2.5898D-04	5.9722D-04	9.3505D-03	1.3833D-02	2.0486D-02	2.5561D-02	3.2957D-02	3.1507D-02	3.0748D-02
14	1.00985	2.2022D-04	4.9825D-04	6.5040D-03	1.0205D-02	1.5561D-02	2.0500D-02	2.8862D-02	3.0413D-02	3.0557D-02
13	0.94112	2.0793D-04	4.6815D-04	5.6751D-03	9.1822D-03	1.4235D-02	1.9012D-02	2.7668D-02	2.9925D-02	2.9946D-02
12	0.87239	2.0414D-04	4.5947D-04	5.4241D-03	8.8983D-03	1.3965D-02	1.8603D-02	2.7673D-02	2.9743D-02	2.9390D-02
11	0.80369	2.1135D-04	4.7724D-04	5.6763D-03	9.3636D-03	1.4802D-02	1.9185D-02	2.8583D-02	2.9475D-02	2.8274D-02
10	0.73500	2.2941D-04	5.2210D-04	6.5024D-03	1.0637D-02	1.6477D-02	2.1412D-02	2.9634D-02	2.9323D-02	2.6925D-02
9	0.66633	2.7779D-04	6.4763D-04	9.2124D-03	1.4691D-02	2.0429D-02	2.8202D-02	3.0348D-02	2.8715D-02	2.4641D-02
8	0.59769	3.7932D-04	9.1624D-04	1.4889D-02	2.2295D-02	2.8385D-02	3.5103D-02	3.0034D-02	2.6639D-02	2.1312D-02
7	0.52908	5.5149D-04	1.3277D-03	2.3957D-02	3.0426D-02	3.5169D-02	3.6314D-02	2.9495D-02	2.4500D-02	1.8960D-02
6	0.46051	8.1213D-04	1.9309D-03	3.3450D-02	3.5594D-02	3.5598D-02	3.3827D-02	2.8481D-02	2.3546D-02	1.9181D-02
5	0.39194	4.8888D-04	1.2909D-03	2.7917D-02	3.0239D-02	3.0709D-02	2.9540D-02	2.6348D-02	2.2735D-02	2.0200D-02
4	0.33203	8.5922D-04	9.8353D-04	1.2071D-03	1.5622D-03	2.2710D-03	3.7887D-03	3.6129D-03	3.0118D-03	2.8415D-03
3	0.28082	7.0451D-04	7.6286D-04	8.6061D-04	9.8953D-04	1.1736D-03	1.4717D-03	1.9449D-03	1.9374D-03	1.7769D-03
2	0.19139	1.9826D-04	1.9798D-04	1.9963D-04	2.0128D-04	2.0292D-04	2.0455D-04	2.0614D-04	2.0763D-04	2.0841D-04
1	0.06378	4.6167D-05	4.5410D-05	4.4775D-05	4.4129D-05	4.3472D-05	4.2805D-05	4.2128D-05	4.1441D-05	4.0742D-05

NODE	(M)	10	11
24	1.70318	2.3138D-05	2.2865D-05
23	1.61805	6.4298D-05	6.3142D-05
22	1.53285	2.0132D-04	1.9624D-04
21	1.46462	1.5807D-03	1.4869D-03
20	1.41335	2.3654D-03	2.2633D-03
19	1.35332	2.2383D-02	2.0387D-02
18	1.28463	2.2599D-02	1.8252D-02
17	1.21598	2.2449D-02	1.5616D-02
16	1.14730	2.3655D-02	1.5589D-02
15	1.07858	2.4585D-02	1.5525D-02
14	1.00985	2.4210D-02	1.4722D-02
13	0.94112	2.3137D-02	1.3645D-02
12	0.87239	2.2035D-02	1.2645D-02
11	0.80369	2.0390D-02	1.1435D-02
10	0.73500	1.8687D-02	1.0322D-02
9	0.66633	1.6514D-02	9.1471D-03
8	0.59769	1.3996D-02	7.9012D-03
7	0.52908	1.2588D-02	7.3845D-03
6	0.46051	1.4058D-02	9.2914D-03
5	0.39194	1.7023D-02	1.3146D-02
4	0.33203	2.6718D-03	2.5021D-03
3	0.28082	1.5216D-03	1.4268D-03
2	0.19139	2.0290D-04	1.9459D-04
1	0.06378	4.0018D-05	3.9153D-05

Figure 8.6.9: DEFORM-4 Fission-gas induced Porosity Fraction Map

CHANNEL 1 SAS4A 3.0 DEFORM-4 TEST PROBLEM FOR SAS4A/SASSYS-1 VERSION 3.0 01/ 9/94 18:20:20 PAGE 125
JOB: 21122/pepper ARGONNE NATIONAL LABORATORY, JANUARY, 1994 USER: 22157/cahalan

*** STEADY STATE TIME (SEC,HRS) = 5.04576D+07 1.40160D+04 ***

NODE	ELEVATION (M)	GRAIN SIZE DISTRIBUTION - DGR2 - M 1	2	3	4	5	6	7	8	9
24	1.70318	1.0000D-05	1.0000D-05	1.0000D-05	1.0000D-05	1.0000D-05	1.0000D-05	1.0000D-05	1.0000D-05	1.0000D-05
23	1.61805	1.0000D-05	1.0000D-05	1.0000D-05	1.0000D-05	1.0000D-05	1.0000D-05	1.0000D-05	1.0000D-05	1.0000D-05
22	1.53285	1.0000D-05	1.0000D-05	1.0000D-05	1.0000D-05	1.0000D-05	1.0000D-05	1.0000D-05	1.0000D-05	1.0000D-05
21	1.46462	1.0007D-05	1.0005D-05	1.0004D-05	1.0003D-05	1.0002D-05	1.0001D-05	1.0001D-05	1.0001D-05	1.0000D-05
20	1.41335	1.0142D-05	1.0096D-05	1.0064D-05	1.0043D-05	1.0028D-05	1.0019D-05	1.0012D-05	1.0008D-05	1.0005D-05
19	1.35332	1.0398D-04	8.7797D-05	7.1261D-05	5.6385D-05	4.3103D-05	3.1481D-05	2.1956D-05	1.5131D-05	1.1432D-05
18	1.28463	1.1189D-04	9.5018D-05	7.6015D-05	5.8401D-05	4.2774D-05	2.9275D-05	1.8742D-05	1.2416D-05	1.0338D-05
17	1.21598	1.2095D-04	1.0357D-04	8.2085D-05	6.1775D-05	4.3990D-05	2.8777D-05	1.7259D-05	1.1353D-05	1.0112D-05
16	1.14730	1.2962D-04	1.1142D-04	8.7747D-05	6.5235D-05	4.5772D-05	2.9277D-05	1.6886D-05	1.1011D-05	1.0061D-05
15	1.07858	1.4097D-04	1.2138D-04	9.5086D-05	7.0016D-05	4.8613D-05	3.0711D-05	1.7167D-05	1.0881D-05	1.0039D-05
14	1.00985	1.4998D-04	1.2910D-04	1.0066D-04	7.3568D-05	5.0630D-05	3.1711D-05	1.7380D-05	1.0800D-05	1.0028D-05
13	0.94112	1.5193D-04	1.3049D-04	1.0131D-04	7.3597D-05	5.0260D-05	3.1125D-05	1.6832D-05	1.0649D-05	1.0020D-05
12	0.87239	1.5123D-04	1.2953D-04	1.0014D-04	7.2343D-05	4.9038D-05	2.9982D-05	1.6040D-05	1.0497D-05	1.0014D-05
11	0.80369	1.4525D-04	1.2375D-04	9.5113D-05	6.8264D-05	4.5820D-05	2.7455D-05	1.4631D-05	1.0315D-05	1.0009D-05
10	0.73500	1.3701D-04	1.1598D-04	8.8612D-05	6.3192D-05	4.1920D-05	2.4571D-05	1.3282D-05	1.0193D-05	1.0006D-05
9	0.66633	1.2407D-04	1.0399D-04	7.8982D-05	5.5994D-05	3.6562D-05	2.1025D-05	1.2050D-05	1.0118D-05	1.0004D-05
8	0.59769	1.1167D-04	9.2083D-05	6.9446D-05	4.8887D-05	3.1417D-05	1.8099D-05	1.1335D-05	1.0085D-05	1.0004D-05
7	0.52908	1.0378D-04	8.4059D-05	6.2974D-05	4.4125D-05	2.8264D-05	1.6654D-05	1.1108D-05	1.0083D-05	1.0004D-05
6	0.46051	9.4469D-05	7.4785D-05	5.5771D-05	3.9188D-05	2.5572D-05	1.5891D-05	1.1173D-05	1.0122D-05	1.0009D-05
5	0.39194	8.4910D-05	6.5899D-05	4.9290D-05	3.5216D-05	2.3937D-05	1.5901D-05	1.1565D-05	1.0248D-05	1.0030D-05
4	0.33203	1.0067D-05	1.0040D-05	1.0024D-05	1.0014D-05	1.0008D-05	1.0005D-05	1.0003D-05	1.0002D-05	1.0001D-05
3	0.28082	1.0004D-05	1.0002D-05	1.0002D-05	1.0001D-05	1.0001D-05	1.0000D-05	1.0000D-05	1.0000D-05	1.0000D-05
2	0.19139	1.0000D-05	1.0000D-05	1.0000D-05	1.0000D-05	1.0000D-05	1.0000D-05	1.0000D-05	1.0000D-05	1.0000D-05
1	0.06378	1.0000D-05	1.0000D-05	1.0000D-05	1.0000D-05	1.0000D-05	1.0000D-05	1.0000D-05	1.0000D-05	1.0000D-05

NODE	(M)	10	11
24	1.70318	1.0000D-05	1.0000D-05
23	1.61805	1.0000D-05	1.0000D-05
22	1.53285	1.0000D-05	1.0000D-05
21	1.46462	1.0000D-05	1.0000D-05
20	1.41335	1.0003D-05	1.0002D-05
19	1.35332	1.0255D-05	1.0035D-05
18	1.28463	1.0032D-05	1.0002D-05
17	1.21598	1.0006D-05	1.0000D-05
16	1.14730	1.0003D-05	1.0000D-05
15	1.07858	1.0001D-05	1.0000D-05
14	1.00985	1.0001D-05	1.0000D-05
13	0.94112	1.0000D-05	1.0000D-05
12	0.87239	1.0000D-05	1.0000D-05
11	0.80369	1.0000D-05	1.0000D-05
10	0.73500	1.0000D-05	1.0000D-05
9	0.66633	1.0000D-05	1.0000D-05
8	0.59769	1.0000D-05	1.0000D-05
7	0.52908	1.0000D-05	1.0000D-05
6	0.46051	1.0001D-05	1.0000D-05
5	0.39194	1.0003D-05	1.0000D-05
4	0.33203	1.0001D-05	1.0000D-05
3	0.28082	1.0000D-05	1.0000D-05
2	0.19139	1.0000D-05	1.0000D-05
1	0.06378	1.0000D-05	1.0000D-05

Figure 8.6.10: DEFORM-4 Fuel Grain Size Map

CHANNEL 1 SAS4A 3.0 DEFORM-4 TEST PROBLEM FOR SAS4A/SASSYS-1 VERSION 3.0 01/ 9/94 18:20:20 PAGE 126
JOB: 21122/pepper ARGONNE NATIONAL LABORATORY, JANUARY, 1994 USER: 22157/cahalan

*** STEADY STATE TIME (SEC,HRS) = 5.04576D+07 1.40160D+04 ***

	ELEVATION	CRACK VOLUME FRACTION								
NODE	(M)	1	2	3	4	5	6	7	8	9
24	1.70318	0.0000D+00	0.0000D+00	0.0000D+00	0.0000D+00	0.0000D+00	0.0000D+00	0.0000D+00	0.0000D+00	0.0000D+00
23	1.61805	0.0000D+00	0.0000D+00	0.0000D+00	0.0000D+00	0.0000D+00	0.0000D+00	0.0000D+00	0.0000D+00	0.0000D+00
22	1.53285	0.0000D+00	0.0000D+00	0.0000D+00	0.0000D+00	0.0000D+00	0.0000D+00	0.0000D+00	0.0000D+00	0.0000D+00
21	1.46462	0.0000D+00	0.0000D+00	0.0000D+00	0.0000D+00	0.0000D+00	0.0000D+00	0.0000D+00	0.0000D+00	0.0000D+00
20	1.41335	0.0000D+00	0.0000D+00	0.0000D+00	0.0000D+00	0.0000D+00	0.0000D+00	0.0000D+00	8.9034D-05	1.3762D-04
19	1.35332	0.0000D+00	8.4941D-06	7.3467D-05	9.0068D-05	7.8064D-05	5.4891D-05	3.8875D-05	3.2034D-05	2.9686D-05
18	1.28463	0.0000D+00	1.8955D-04	6.8431D-04	8.6019D-04	8.7192D-04	7.9414D-04	6.9761D-04	6.2069D-04	1.1111D-03
17	1.21598	0.0000D+00	0.0000D+00	2.0019D-04	3.7953D-04	3.9626D-04	3.0242D-04	2.1719D-04	5.8894D-04	1.4921D-03
16	1.14730	0.0000D+00	0.0000D+00	0.0000D+00	0.0000D+00	0.0000D+00	0.0000D+00	0.0000D+00	0.0000D+00	1.0478D-03
15	1.07858	0.0000D+00	0.0000D+00	0.0000D+00	0.0000D+00	0.0000D+00	0.0000D+00	0.0000D+00	0.0000D+00	1.0203D-03
14	1.00985	0.0000D+00	0.0000D+00	0.0000D+00	0.0000D+00	0.0000D+00	0.0000D+00	0.0000D+00	0.0000D+00	1.1713D-03
13	0.94112	0.0000D+00	0.0000D+00	0.0000D+00	0.0000D+00	0.0000D+00	0.0000D+00	0.0000D+00	0.0000D+00	1.2699D-03
12	0.87239	0.0000D+00	0.0000D+00	0.0000D+00	0.0000D+00	0.0000D+00	0.0000D+00	0.0000D+00	0.0000D+00	7.9978D-04
11	0.80369	0.0000D+00	0.0000D+00	0.0000D+00	0.0000D+00	0.0000D+00	0.0000D+00	0.0000D+00	0.0000D+00	9.2114D-04
10	0.73500	0.0000D+00	0.0000D+00	0.0000D+00	0.0000D+00	0.0000D+00	0.0000D+00	0.0000D+00	3.6966D-04	1.5801D-03
9	0.66633	0.0000D+00	0.0000D+00	0.0000D+00	0.0000D+00	0.0000D+00	0.0000D+00	0.0000D+00	5.5004D-04	1.6250D-03
8	0.59769	0.0000D+00	0.0000D+00	0.0000D+00	0.0000D+00	0.0000D+00	0.0000D+00	0.0000D+00	7.1546D-04	1.6236D-03
7	0.52908	0.0000D+00	0.0000D+00	0.0000D+00	4.1849D-05	8.0314D-05	5.3133D-05	7.0643D-04	1.5155D-03	2.2349D-03
6	0.46051	0.0000D+00	0.0000D+00	5.6047D-04	8.1323D-04	8.3952D-04	7.8326D-04	8.0652D-04	1.4137D-03	2.2897D-03
5	0.39194	0.0000D+00	2.5679D-05	3.5952D-04	4.4061D-04	4.4336D-04	4.2807D-04	4.1031D-04	7.2176D-04	1.2089D-03
4	0.33203	0.0000D+00	0.0000D+00	0.0000D+00	0.0000D+00	0.0000D+00	0.0000D+00	4.4057D-04	4.0202D-04	2.8381D-04
3	0.28082	0.0000D+00	0.0000D+00	0.0000D+00	0.0000D+00	0.0000D+00	0.0000D+00	0.0000D+00	0.0000D+00	4.0847D-04
2	0.19139	0.0000D+00	0.0000D+00	0.0000D+00	0.0000D+00	0.0000D+00	0.0000D+00	0.0000D+00	0.0000D+00	0.0000D+00
1	0.06378	0.0000D+00	0.0000D+00	0.0000D+00	0.0000D+00	0.0000D+00	0.0000D+00	0.0000D+00	0.0000D+00	0.0000D+00

NODE	(M)	10	11
24	1.70318	0.0000D+00	0.0000D+00
23	1.61805	0.0000D+00	0.0000D+00
22	1.53285	0.0000D+00	0.0000D+00
21	1.46462	0.0000D+00	1.5966D-05
20	1.41335	8.1658D-05	9.1073D-05
19	1.35332	2.2579D-04	8.6337D-04
18	1.28463	1.7628D-03	2.4292D-03
17	1.21598	2.2907D-03	3.0595D-03
16	1.14730	1.9966D-03	2.8695D-03
15	1.07858	2.0898D-03	3.0441D-03
14	1.00985	2.3022D-03	3.2930D-03
13	0.94112	2.4027D-03	3.3879D-03
12	0.87239	1.9501D-03	2.9396D-03
11	0.80369	2.0219D-03	2.9716D-03
10	0.73500	2.5833D-03	3.4627D-03
9	0.66633	2.5219D-03	3.3253D-03
8	0.59769	2.3936D-03	3.1052D-03
7	0.52908	2.8592D-03	3.4644D-03
6	0.46051	2.7829D-03	3.2901D-03
5	0.39194	1.9365D-03	2.4460D-03
4	0.33203	1.9989D-04	1.2609D-04
3	0.28082	3.9787D-04	4.0593D-04
2	0.19139	0.0000D+00	0.0000D+00
1	0.06378	0.0000D+00	0.0000D+00

Figure 8.6.11: DEFORM-4 Fuel Crack Volume Fraction Map

CHANNEL 1 SAS4A 3.0 DEFORM-4 TEST PROBLEM FOR SAS4A/SASSYS-1 VERSION 3.0 01/ 9/94 18:20:20 PAGE 127
JOB: 21122/pepper ARGONNE NATIONAL LABORATORY, JANUARY, 1994 USER: 22157/cahalan

*** STEADY STATE TIME (SEC,HRS) = 5.04576D+07 1.40160D+04 ***

	ELEVATION	RADIAL MESH POINTS - R - M								
NODE	(M)	1	2	3	4	5	6	7	8	9
24	1.70318	1.0027D-06	6.7091D-04	1.1620D-03	1.5002D-03	1.7750D-03	2.0127D-03	2.2251D-03	2.4189D-03	2.5983D-03
23	1.61805	1.0032D-06	6.7136D-04	1.1628D-03	1.5012D-03	1.7762D-03	2.0140D-03	2.2265D-03	2.4204D-03	2.5999D-03
22	1.53285	1.0040D-06	6.7207D-04	1.1640D-03	1.5027D-03	1.7780D-03	2.0160D-03	2.2287D-03	2.4228D-03	2.6025D-03
21	1.46462	1.0056D-06	6.7411D-04	1.1675D-03	1.5072D-03	1.7833D-03	2.0220D-03	2.2354D-03	2.4301D-03	2.6103D-03
20	1.41335	1.0120D-06	6.7534D-04	1.1697D-03	1.5100D-03	1.7867D-03	2.0262D-03	2.2402D-03	2.4354D-03	2.6160D-03
19	1.35332	6.2416D-04	9.0732D-04	1.3044D-03	1.6022D-03	1.8543D-03	2.0803D-03	2.2904D-03	2.4876D-03	2.6717D-03
18	1.28463	6.8944D-04	9.4706D-04	1.3210D-03	1.6143D-03	1.8629D-03	2.0864D-03	2.2963D-03	2.4945D-03	2.6791D-03
17	1.21598	7.1117D-04	9.6418D-04	1.3323D-03	1.6236D-03	1.8708D-03	2.0915D-03	2.3003D-03	2.4994D-03	2.6844D-03
16	1.14730	7.2223D-04	9.7330D-04	1.3400D-03	1.6282D-03	1.8733D-03	2.0911D-03	2.2978D-03	2.4973D-03	2.6829D-03
15	1.07858	7.3297D-04	9.8205D-04	1.3473D-03	1.6336D-03	1.8772D-03	2.0932D-03	2.2960D-03	2.4949D-03	2.6810D-03
14	1.00985	7.3975D-04	9.8751D-04	1.3519D-03	1.6371D-03	1.8798D-03	2.0949D-03	2.2950D-03	2.4930D-03	2.6792D-03
13	0.94112	7.3792D-04	9.8621D-04	1.3510D-03	1.6362D-03	1.8788D-03	2.0938D-03	2.2939D-03	2.4921D-03	2.6781D-03
12	0.87239	7.3307D-04	9.8257D-04	1.3481D-03	1.6337D-03	1.8765D-03	2.0916D-03	2.2927D-03	2.4913D-03	2.6775D-03
11	0.80369	7.2104D-04	9.7346D-04	1.3413D-03	1.6279D-03	1.8713D-03	2.0868D-03	2.2911D-03	2.4909D-03	2.6770D-03
10	0.73500	7.0705D-04	9.6287D-04	1.3333D-03	1.6212D-03	1.8654D-03	2.0817D-03	2.2903D-03	2.4908D-03	2.6766D-03
9	0.66633	6.8141D-04	9.4357D-04	1.3186D-03	1.6091D-03	1.8551D-03	2.0762D-03	2.2904D-03	2.4911D-03	2.6765D-03
8	0.59769	6.5147D-04	9.2115D-04	1.3014D-03	1.5953D-03	1.8446D-03	2.0738D-03	2.2919D-03	2.4922D-03	2.6768D-03
7	0.52908	6.1476D-04	8.9485D-04	1.2852D-03	1.5838D-03	1.8392D-03	2.0744D-03	2.2933D-03	2.4929D-03	2.6766D-03
6	0.46051	5.2819D-04	8.4443D-04	1.2595D-03	1.5702D-03	1.8344D-03	2.0740D-03	2.2926D-03	2.4916D-03	2.6748D-03
5	0.39194	3.9737D-04	7.8756D-04	1.2331D-03	1.5554D-03	1.8269D-03	2.0687D-03	2.2870D-03	2.4860D-03	2.6692D-03
4	0.33203	1.0168D-06	6.7549D-04	1.1699D-03	1.5102D-03	1.7868D-03	2.0261D-03	2.2401D-03	2.4353D-03	2.6158D-03
3	0.28082	1.0065D-06	6.7437D-04	1.1680D-03	1.5077D-03	1.7839D-03	2.0226D-03	2.2360D-03	2.4308D-03	2.6110D-03
2	0.19139	1.0036D-06	6.7191D-04	1.1637D-03	1.5023D-03	1.7775D-03	2.0155D-03	2.2281D-03	2.4221D-03	2.6017D-03
1	0.06378	1.0023D-06	6.7079D-04	1.1618D-03	1.4999D-03	1.7746D-03	2.0122D-03	2.2246D-03	2.4183D-03	2.5976D-03

NODE	(M)	10	11	12	R(NE,J)	R(NEP,J)	IEQCOL	IUNEQA
24	1.70318	2.7661D-03	2.9242D-03	3.0002D-03	3.0903D-03	3.6544D-03	1	1
23	1.61805	2.7678D-03	2.9260D-03	3.0020D-03	3.0903D-03	3.6544D-03	1	1
22	1.53285	2.7705D-03	2.9289D-03	3.0049D-03	3.0903D-03	3.6544D-03	1	1
21	1.46462	2.7788D-03	2.9376D-03	3.0138D-03	3.0903D-03	3.6544D-03	1	1
20	1.41335	2.7847D-03	2.9438D-03	3.0202D-03	3.0903D-03	3.6544D-03	1	1
19	1.35332	2.8439D-03	3.0057D-03	3.0832D-03	3.0907D-03	3.6549D-03	6	9
18	1.28463	2.8511D-03	3.0127D-03	3.0899D-03	3.0904D-03	3.6545D-03	6	9
17	1.21598	2.8568D-03	3.0183D-03	3.0953D-03	3.0953D-03	3.6586D-03	7	8
16	1.14730	2.8560D-03	3.0181D-03	3.0952D-03	3.0952D-03	3.6584D-03	7	8
15	1.07858	2.8547D-03	3.0173D-03	3.0946D-03	3.0946D-03	3.6577D-03	7	8
14	1.00985	2.8531D-03	3.0158D-03	3.0932D-03	3.0932D-03	3.6564D-03	7	8
13	0.94112	2.8520D-03	3.0145D-03	3.0917D-03	3.0917D-03	3.6549D-03	7	8
12	0.87239	2.8512D-03	3.0135D-03	3.0907D-03	3.0907D-03	3.6538D-03	7	8
11	0.80369	2.8505D-03	3.0126D-03	3.0897D-03	3.0897D-03	3.6526D-03	7	8
10	0.73500	2.8498D-03	3.0114D-03	3.0884D-03	3.0884D-03	3.6512D-03	6	8
9	0.66633	2.8489D-03	3.0100D-03	3.0868D-03	3.0868D-03	3.6494D-03	6	8
8	0.59769	2.8484D-03	3.0088D-03	3.0854D-03	3.0854D-03	3.6479D-03	6	7
7	0.52908	2.8475D-03	3.0074D-03	3.0838D-03	3.0838D-03	3.6462D-03	5	7
6	0.46051	2.8453D-03	3.0052D-03	3.0816D-03	3.0816D-03	3.6440D-03	5	7
5	0.39194	2.8400D-03	3.0003D-03	3.0770D-03	3.0802D-03	3.6425D-03	4	7
4	0.33203	2.7846D-03	2.9437D-03	3.0200D-03	3.0792D-03	3.6413D-03	1	1
3	0.28082	2.7794D-03	2.9382D-03	3.0144D-03	3.0790D-03	3.6411D-03	1	1
2	0.19139	2.7696D-03	2.9279D-03	3.0040D-03	3.0789D-03	3.6409D-03	1	1
1	0.06378	2.7654D-03	2.9235D-03	2.9994D-03	3.0788D-03	3.6408D-03	1	1

Figure 8.6.12: DEFORM-4 Radial Node Locations

CHANNEL 1 SAS4A 3.0 DEFORM-4 TEST PROBLEM FOR SAS4A/SASSYS-1 VERSION 3.0 01/ 9/94 18:20:20 PAGE 128
JOB: 21122/pepper ARGONNE NATIONAL LABORATORY, JANUARY, 1994 USER: 22157/cahalan

*** STEADY STATE TIME (SEC,HRS) = 0.00000D+00 0.00000D+00 * 1

							********** FAILURE CRITERIA **********								
AXIAL FUEL NODE	AXIAL ELEV (M)	FUEL/CLAD GAP (M)	F/C PRESS (MPA)	HOOP STRESS (MPA)	FLOW STRESS (MPA)	CLAD STRAIN PERCENT	K-D FAIL STRN	NSMH LIFE FRAC	NSMH BRST PRES	NSMH FAIL TEMP	INPUT STRAN LIMIT	INPUT MFAIL LIMIT	HEDL LMP FRAC	HEDL 111 FRAC	HEDL DORN FRAC	TDC2 LIFE FRAC
24	1.7032	9.0104D-05	2.63	13.30	624.10	0.0000	0.0000	0.0000	0.0279	0.5363	0.0000	0.0000	0.0007	0.0007	0.0000	0.0001
23	1.6180	8.8267D-05	2.63	13.20	624.00	0.0000	0.0000	0.0000	0.0279	0.5362	0.0000	0.0000	0.0007	0.0007	0.0000	0.0001
22	1.5329	8.5360D-05	2.63	13.10	623.83	0.0000	0.0000	0.0000	0.0279	0.5361	0.0000	0.0000	0.0007	0.0007	0.0000	0.0001
21	1.4646	7.6460D-05	2.63	13.02	621.84	0.0000	0.0000	0.0000	0.0279	0.5364	0.0000	0.0000	0.0008	0.0008	0.0000	0.0001
20	1.4133	7.0105D-05	2.63	12.96	621.50	0.0000	0.0000	0.0000	0.0279	0.5362	0.0000	0.0000	0.0007	0.0007	0.0000	0.0001
19	1.3533	7.5047D-06	2.63	12.88	597.88	0.0000	0.0000	0.0000	0.0281	0.5410	0.0000	0.0000	0.0013	0.0013	0.0000	0.0002
18	1.2846	4.9918D-07	2.63	12.80	602.77	0.0000	0.0000	0.0000	0.0279	0.5376	0.0000	0.0000	0.0009	0.0009	0.0000	0.0001
17	1.2160	0.0000D+00	8.68	49.26	611.13	0.0000	0.0000	0.0000	0.0913	0.5328	0.0000	0.0000	0.0005	0.0005	0.0002	0.0007
16	1.1473	0.0000D+00	13.80	80.08	624.37	0.0002	0.0000	0.0000	0.1438	0.5264	0.0000	0.0000	0.0002	0.0002	0.0007	0.0018
15	1.0786	0.0000D+00	20.66	121.42	636.73	0.0008	0.0000	0.0000	0.2132	0.5194	0.0000	0.0000	0.0001	0.0001	0.0019	0.0040
14	1.0098	0.0000D+00	28.50	168.59	640.70	0.0020	0.0000	0.0000	0.2911	0.5115	0.0000	0.0000	0.0001	0.0001	0.0035	0.0065
13	0.9411	0.0000D+00	33.26	197.17	645.37	0.0016	0.0000	0.0000	0.3362	0.5027	0.0000	0.0000	0.0001	0.0001	0.0028	0.0050
12	0.8724	0.0000D+00	35.87	212.78	650.31	0.0008	0.0000	0.0000	0.3589	0.4936	0.0000	0.0000	0.0000	0.0000	0.0014	0.0026
11	0.8037	0.0000D+00	36.57	216.91	655.54	0.0003	0.0000	0.0000	0.3626	0.4841	0.0000	0.0000	0.0000	0.0000	0.0005	0.0009
10	0.7350	0.0000D+00	35.25	208.80	660.89	0.0001	0.0000	0.0000	0.3467	0.4745	0.0000	0.0000	0.0000	0.0000	0.0001	0.0002
9	0.6663	0.0000D+00	29.66	175.00	666.55	0.0000	0.0000	0.0000	0.2897	0.4648	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
8	0.5977	0.0000D+00	23.89	140.07	672.19	0.0000	0.0000	0.0000	0.2318	0.4552	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
7	0.5291	0.0000D+00	19.57	113.96	676.73	0.0000	0.0000	0.0000	0.1889	0.4464	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
6	0.4605	0.0000D+00	8.68	48.24	679.48	0.0000	0.0000	0.0000	0.0833	0.4376	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
5	0.3919	3.1913D-06	2.63	11.73	682.01	0.0000	0.0000	0.0000	0.0251	0.4296	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
4	0.3320	5.9128D-05	2.63	11.66	686.59	0.0000	0.0000	0.0000	0.0250	0.4184	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
3	0.2808	6.4626D-05	2.63	11.60	687.07	0.0000	0.0000	0.0000	0.0249	0.4170	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
2	0.1914	7.4914D-05	2.63	11.49	687.77	0.0000	0.0000	0.0000	0.0249	0.4151	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
1	0.0638	7.9379D-05	2.63	11.34	688.18	0.0000	0.0000	0.0000	0.0249	0.4139	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000

GAS PLENUM PRESSURE	=	2.6343D+06	PA
MOLTEN CAVITY PRESSURE	=	0.0000D+00	PA
TOTAL FISSION GAS PRODUCED	=	2.7084D-03	KG
FISSION GAS IN SOLID FUEL	=	7.6230D-04	KG
FISSION GAS IN MOLTEN FUEL	=	0.0000D+00	KG
FRACTIONAL GAS RELEASE	=	0.71853743	
MOLTEN CAVITY VOLUME	=	0.0000D+00	M3
MOLTEN CAVITY GAS CONTENT	=	0.0000D+00	MOL
MOLTEN CAVITY TEMPERATURE	=	0.0000D+00	K
MASS MOLTEN FUEL IN PIN	=	0.0000D+00	KG
MASS SOLID FUEL IN PIN	=	4.6439D-01	KG
PLENUM HELIUM GAS CONTENT	=	1.2426D-03	MOL
PLENUM FISSION GAS CONTENT	=	1.4691D-02	MOL
AXIAL EXPANSION REACTIVITY	=	0.0000D+00	DOL

Figure 8.6.13: DEFORM-4 Short Form Pin Summary Output

Short summary output from DEFORM-4 showing the values calculated for various failure parameters. The hoop stress values are a simple average based on the inner and outer pressures and thickness of the clad.

If a failure criterion has a value greater than 1.0, then that criterion predicts failure. Currently, only the MFAIL parameters are used to automatically initiate failure of the pin.

The amount of output can be controlled through the use of NSKIP and IPRD in block 51 of the input. In the transient state the output from DEFORM-4 is printed at the end of a heat-transfer time step, as controlled by IPO and IPOBOI.

Listing 8.6.1: Sample DEFORM-4 Output using 15-15Ti Model (**IDEFOPT** > 0 and **ICTYPE** = 4)

```
 AXIAL  AXIAL    BURNUP   CLAD     CLAD          CLAD       CLAD       CLAD       ␣
↪CLAD       FUEL PINS
 FUEL   ELEV     ATOM     DOSE     OUTER        SWELLING  IRRADIATION THERMAL    ␣
↪FAILURE    FAILURE
 NODE    (M)      (%)     (DPA)   CORROSION(M)   (%)       CREEP(%)   CREEP(%)   ␣
↪MARGIN     FRACTION
  24   1.1940   0.0566   0.4579  0.000E+00    0.000E+00  0.388E-03   0.127E-02  ␣
↪0.100E-05  0.000E+00
  23   1.1840   0.0566   0.4579  0.000E+00    0.000E+00  0.388E-03   0.127E-02  ␣
↪0.100E-05  0.000E+00
  22   1.1551   2.2660  18.1978  0.000E+00    0.000E+00  0.151E-01   0.136E-02  ␣
↪0.100E-05  0.000E+00
  21   1.1069   6.8080  53.7328  0.000E+00    0.000E+00  0.438E-01   0.152E-02  ␣
↪0.100E-05  0.000E+00
  20   1.0582   7.8651  61.9464  0.000E+00    0.000E+00  0.514E-01   0.138E-02  ␣
↪0.100E-05  0.000E+00
  19   1.0094   8.8115  69.3621  0.000E+00    0.000E+00  0.581E-01   0.123E-02  ␣
↪0.100E-05  0.000E+00
  18   0.9607   9.7293  76.5560  0.000E+00    0.682E+00  0.636E-01   0.111E-02  ␣
↪0.100E-05  0.000E+00
  17   0.9118  10.4418  82.1941  0.000E+00    0.127E+01  0.654E-01   0.990E-03  ␣
↪0.100E-05  0.000E+00
  16   0.8629  11.0819  87.1732  0.000E+00    0.178E+01  0.707E-01   0.917E-03  ␣
↪0.100E-05  0.000E+00
  15   0.8140  11.4866  90.2900  0.000E+00    0.234E+01  0.990E-01   0.953E-03  ␣
↪0.100E-05  0.000E+00
  14   0.7651  11.7833  92.7017  0.000E+00    0.280E+01  0.138E+00   0.877E-03  ␣
↪0.100E-05  0.000E+00
  13   0.7163  11.9357  94.0680  0.000E+00    0.308E+01  0.194E+00   0.840E-03  ␣
↪0.100E-05  0.000E+00
  12   0.6676  11.9144  93.6432  0.000E+00    0.329E+01  0.286E+00   0.105E-02  ␣
↪0.100E-05  0.000E+00
  11   0.6188  11.7873  92.6803  0.000E+00    0.386E+01  0.369E+00   0.121E-02  ␣
↪0.100E-05  0.000E+00
  10   0.5699  11.4915  90.4318  0.000E+00    0.413E+01  0.408E+00   0.117E-02  ␣
↪0.100E-05  0.000E+00
   9   0.5210  11.1543  87.7450  0.000E+00    0.447E+01  0.477E+00   0.150E-02  ␣
↪0.100E-05  0.000E+00
   8   0.4721  10.5050  82.6725  0.000E+00    0.378E+01  0.514E+00   0.201E-02  ␣
↪0.100E-05  0.000E+00
   7   0.4230   9.8220  77.2449  0.000E+00    0.303E+01  0.733E+00   0.159E+00  ␣
↪0.100E-05  0.000E+00
   6   0.3740   9.0097  70.9643  0.000E+00    0.190E+01  0.489E+00   0.418E-02  ␣
↪0.100E-05  0.000E+00
```

(continues on next page)

(continued from previous page)

```
   5   0.3252   8.0608  63.5630  0.000E+00    0.103E+01  0.456E+00   0.645E-02 ␣
↪0.100E-05  0.000E+00
   4   0.2633   0.0076   0.0612  0.000E+00    0.000E+00  0.459E-03   0.198E-01 ␣
↪0.100E-05  0.000E+00
   3   0.1880   0.0076   0.0612  0.000E+00    0.000E+00  0.459E-03   0.198E-01 ␣
↪0.100E-05  0.000E+00
   2   0.1128   0.0076   0.0612  0.000E+00    0.000E+00  0.459E-03   0.199E-01 ␣
↪0.100E-05  0.000E+00
   1   0.0376   0.0076   0.0612  0.000E+00    0.000E+00  0.459E-03   0.199E-01 ␣
↪0.100E-05  0.000E+00
```

Listing 8.6.1 shows an example output from a fuel performance with 15-15Ti cladding in a fuel channel. Columns show the axial node number, the axial elevation (m), fuel burnup (at%), clad dose (dpa), clad outer corrosion thickness (m), which is controlled by **COCOR** input, irradiation induced volumetric clad swelling (%), clad irradiation and thermal creep hoop strain (%), clad failure margin due to thermal creep rupture, which is the computed Clad Damage Fraction value (CDF) and fuel pin failure fraction (or probability) computed if stochastic clad damage evaluation module is switched ON (**IDEFSTFAL** > 0 and **IDEFOPT** > 0). Irradiation creep, thermal creep, void swelling, and clad creep rupture and clad failure fractions are computed by EPSIR15Ti, ET15Ti, EPSWC15Ti, CRUP15Ti, and CDFNORMALDIST routines, respectively. This output can be printed together with the DEFORM-4 output for normal operation and transient simulations at user specified steps.

8.7 Material Properties

All material property data are cast as functions or subroutines to allow for modularization and the ease of making changes. This also allows for the incorporation of different materials data in a straightforward manner. In a number of the correlations used, the units are inconsistent with the SI unit system adopted by SAS4A. The routines that use these correlations carry out appropriate units conversions internally.

8.7.1 Fuel Density

The solid fuel density is assumed to have the functional form

$$\rho_f = \frac{\rho_o}{1 + C_1 (T - 273) + C_2 (T - 273)^2} \tag{8.7-1}$$

where

ρ_o = The theoretical density at 273 K, kg/m^3

C_1, C_2 = Input coefficients

T = Temperature, K

This applies between 273 K and the solidus temperature.

The liquid fuel density is given by

$$\rho_l = \frac{\rho_o}{1 + C_3 (T - 273)} \tag{8.7-2}$$

where

C_3 = Input coefficient

This applies to temperatures above the liquidus. For the range between the solidus and liquidus temperatures, a linear interpolation is performed.

These equations are found in the function RHOF. Suggested values of coefficients are from the Nuclear Systems Materials Handbook [8-17].

ρ_o = COEFDS (1) = 11.05×10^3 kg/m^3 (mixed oxide)

C_1 = COEFDS (2) = 2.04×10^{-5} K^{-1}

C_2 = COEFDS (3) = 8.70×10^{-9} K^{-2}

C_3 = COEFDL (2) = 9.30×10^{-5} K^{-1}

8.7.2 Fuel Thermal Expansion Coefficient

The thermal expansion coefficient of the fuel is derived from the coefficients of the fuel density function. This assures consistency between the thermal expansion and the density.

The specific volume (inverse of density) at two temperatures is given by

$$v_1 = \frac{1 + C_1 (T_1 - 273) + C_2 (T_1 - 273)^2}{\rho_o} \tag{8.7-3}$$

$$v_2 = \frac{1 + C_1 (T_2 - 273) + C_2 (T_2 - 273)^2}{\rho_o} \tag{8.7-4}$$

The change in volume is therefore given by

$$\begin{aligned} \Delta v_1 = v_2 - v_1 = \frac{1}{\rho_o} \{ & [C_1 + C_2 (T_2 - 273)] (T_2 - 273) \\ & - [C_1 + C_2 (T_1 - 273)] (T_1 - 273) \} \end{aligned} \tag{8.7-5}$$

Eq. (8.7-5) can be rewritten in terms of the bulk thermal expansion coefficient, β.

$$\Delta v = \frac{1}{\rho_o} \{ \beta (T_2) (T_2 - T_r) - \beta (T_1) (T_2 - T_r) \} \tag{8.7-6}$$

where

$\beta (T) = C_1 + C_2 T$

T = Temperature, K

C_1, C_2 = Density function coefficients

T_r = Reference temperature, K

Since the linear expansion coefficient is assumed to be a third of the bulk expansion coefficient, the linear expansion coefficient can be defined from the density coefficients as

$$\alpha (T) + C'_1 + C'_2 T \tag{8.7-7}$$

where

α = Linear thermal expansion coefficient

$C'_1 = \frac{C_1}{3}$

$C'_2 = \frac{C_2}{3}$

T = Temperature, K

This is found in the function ALPHF.

8.7.3 Fuel Modulus of Elasticity

The modulus of elasticity, Young's Modulus, is determined from the bulk modulus by

$$E = 3\left(1 - 2v\right)K \tag{8.7-8}$$

where

E = Modulus of elasticity, Pa

K = Bulk modulus, Pa

v = Poisson's ratio

The bulk modulus is from the NSMH [8-17] and is given by

$$K = 1.38 \times 10^{11}\left[1 - 0.5\left(T/2760\right)^2\right]\left(1 - 2P\right) \tag{8.7-9}$$

where

K = Bulk moduus, Pa

T = Temperature, °C

$P = 1 - \rho_{\text{TD}}$ = Porosity fraction

ρ_{TD} = Fraction of theoretical density

Based on the evaluation in MATPRO-l0 [8-11], if this calculated modulus of elasticity becomes less than 7.9×10^8, it is fixed at this value.

These equations are solved in the function EFUELF.

8.7.4 Mixed Oxide Fuel Thermal Conductivity

Four different options exist for the fuel thermal conductivity. These are controlled through the input parameter IRHOK.

IRHOK = 0

The thermal conductivity as a function of temperature is input in table form through the variable arrays XKTAB and XKTEM.

IRHOK = 1

For this option, the conductivity equations [8-3] are given by

$$k_1\left(T\right) = 1.1 + \frac{1 \times 10^2}{T\left(0.4848 - 0.4465 f_{\text{D}}\right)} \quad \text{for } 800 \text{ C} \le T \le 2000 \text{ C} \tag{8.7-10}$$

$$k_1\left(T\right) = k_1\left(800\right)\frac{168.844}{12.044 + \left(0.196\right)T} \quad \text{for } T < 800 \text{ C} \tag{8.7-11}$$

$$k_3\left(T\right) = k_1\left(2000\right) \quad \text{for } T > 2000 \text{ C} \tag{8.7-12}$$

where

k_1, k_2, k_3 = Fuel thermal conductivity, W/m-K

T = Temperature, °C

f_{D} = Fuel fraction of theoretical density

IRHOK = 2

This form of the conductivity [8-22] is given by

$$k_1(T) = [(C_1 - f_D) f_D - 1] \left[\frac{1}{(C_1 + C_3 T)} + C_4 T^3 \right] \quad \text{for } 0.75 \le f_D \le 0.95 \tag{8.7-13}$$

$$k_2(T) = (3 f_D - 1) \left[\frac{1}{(C_5 + C_6 T)} + C_7 T^3 \right] \quad \text{for } f_D > 0.95 \tag{8.7-14}$$

where

C_1, C_2, C_3, C_4, C_5, C_6, C_7 = Input variables

k_1, k_2 = Fuel conductivity W/m-K

T = Temperature, K

If T is greater than the melting temperature, it is set to the melting temperature.

Suggested values:

$C_1 = \text{COEFK}(1) = 2.1$

$C_2 = \text{COEFK}(2) = 2.88 \times 10^{-3}$

$C_3 = \text{COEFK}(3) = 2.52 \times 10^{-5}$

$C_4 = \text{COEFK}(4) = 5.83 \times 10^{-10}$

$C_5 = \text{COEFK}(5) = 5.75 \times 10^{-2}$

$C_6 = \text{COEFK}(6) = 5.03 \times 10^{-4}$

$C_7 = \text{COEFK}(7) = 2.91 \times 10^{-11}$

IRHOK = 3

This conductivity form is [8-32]

$$k_1(T) = \frac{4.005 \times 10^3}{(T - 273) + 402.4} + 0.6416 \times 10^{-10} T^3 \tag{8.7-15}$$

where

T = Temperature, K

k = Conductivity in W/m-K

This is the correlation for UO_2 and is converted to mixed oxide by subtracting 0.2.

$$k_2(T) = k_1(T) - 0.2 \tag{8.7-16}$$

The porosity correction term was derived for use in the COMETHE-IIIJ [8-33] code and is given by

$$f_p = 1 - 1.029\varepsilon - 3.2\varepsilon^2 - 40.1\varepsilon^3 + 158\varepsilon^4 \tag{8.7-17}$$

where

f_p = Porosity multiplier

$\varepsilon = 1 - \rho_f$ = Fractional porosity

ρ_f = Fractional fuel density = actual density/theoretical density

The conductivity is therefore given by

$$k\left(T\right) = f_{\mathrm{p}} \times k_2\left(T\right) \tag{8.7-18}$$

Two different routines contain the above correlations, FK and KFUEL. The function FK returns a single value of the conductivity for a single invocation and is used in the pre-transient. The subroutine KFUEL returns the conductivity values for each radial node in the current axial segment. It is used in the transient calculational procedure.

8.7.5 Mixed Oxide Fuel Fracture Strength

The fracture strength for mixed-oxide fuel is from the NSMH [8-17] and is for unirradiated fuel.

$$\sigma_{\mathrm{f}} = 2.74 \times 10^7 + 5.9 \times 10^4 T \tag{8.7-19}$$

This is coded in the function SIGFRA.

8.7.6 Mixed Oxide Fuel Creep Rate

The fuel creep rate function is from the NSMH [8-17] and is used in the subroutine FSWELL. It represents diffusional flow, dislocation creep, fission-enhanced thermal creep, and fission-induced creep.

$$\begin{aligned}\dot{\varepsilon}_{\text{total}} = {} & \frac{A}{d^2}\left[1 + 2.11\left(97 - \text{TD}\right)\right]\sigma \exp\left(-\frac{Q_1}{\text{RT}}\right) \\ & + B\left[1 + 0.22\left(97 - \text{TD}\right)\right]\sigma^{4.4} \exp\left(-\frac{Q_2}{\text{RT}}\right) \\ & + C\sigma\dot{F} \exp\left(-\frac{Q_3}{\text{RT}}\right) + D\sigma\dot{F}\end{aligned} \tag{8.7-20}$$

where

$\dot{\varepsilon}_{\text{total}}$ = Creep rate, s^{-1}

$A = 8.97222 \times 10^5$

d = Grain size, μm

TD = Fuel percent of theoretical density

σ = Stress, MPa

$Q_1 = 3.87173 \times 10^5$

R = Gas constant = 8.3169 J/mole-K

T = Temperature, K

$\beta = 9.0 \times 10^2$

$Q_2 = 5.72598 \times 10^5$

$C = 7.8889 \times 10^{-21}$

F = Fission rate, fissions/cm^3-s

$Q_3 = 5.7343 \times 10^4$

$D = 1.5 \times 10^{-24}$

This is used to calculate the fuel creep for fission-gas bubble expansion or contraction in the routine FSWELL.

8.7.7 Cladding Thermal Expansion Coefficient

The mean thermal expansion coefficient of the cladding is taken from the Nuclear Systems Materials Handbook [8-17]. It has the form

$$\alpha_{\mathrm{m}} = C_1 \left(C_2\right)^{\left(C_3\right)^T} \tag{8.7-21}$$

where

α_{m} = Mean thermal expansion coefficient, °F^{-1}

C_1, C_2, C_3 = Calibration constants

T = Temperature, °F

The values for the calibration are given below:

$C_1 = 11.397$

$C_2 = 0.71828$

$C_3 = 0.99890$

This is calculated in the function ALPHC.

8.7.8 Cladding Modulus of Elasticity

The cladding modulus of elasticity is taken from the NSHM [8-17] and is given by

$$E_{\mathrm{c}} = 2.833669 \times 10^{1} - 2.882211 \times 10^{-3} T - 3.697849 \times 10^{-6} T^2 + 7.709188 \times 10^{-10} T^3 \tag{8.7-22}$$

where

E = Modulus of elasticity, Mpsi

T = Temperature, °F

This is then converted to Pa by multiplying by 6.894757×10^{9}. This correlation is coded in the function ECLADF.

8.7.9 Cladding Ultimate Tensile Strength

The ultimate tensile strength of unirradiated, 20% cold-worked, 316 SS is taken from the NSMH [8-17].

$$\sigma_{\mathrm{u}} = C_0 + C_1 T + C_2 T^2 + C_3 T^3 + C_4 T^4 + C_5 T^5 + C_6 T^6 + C_7 T^7 \tag{8.7-23}$$

where

σ_{u} = Ultimate tensile strength, ksi

T = Temperature, F

$C_0 = 1.220241 \times 10^{2}$

$C_1 = -1.015998 \times 10^{-1}$

$C_2 = 8.336636 \times 10^{-4}$

$C_3 = -3.365737 \times 10^{-6}$

$C_4 = 6.227377 \times 10^{-9}$

$C_5 = -5.736229 \times 10^{-12}$

$C_6 = 2.542064 \times 10^{-15}$

$C_7 = -4.321098 \times 10^{-19}$

This is then converted to SI units by multiplying by 6.894757×10^6.

This correlation is good for temperatures below 1200 K only. For temperatures above 1200 K, the curve is assumed to go to zero at the cladding solidus according to the following equation.

$$\sigma_{\text{u}} = \sigma_{\text{u}}\left(1200K\right)\left\{1 - \left[\frac{T - 1200}{T_{\text{mc}} - 1200}\right]^2\right\} \tag{8.7-24}$$

where

$\sigma_{\text{u}}\left(1200K\right) = 1.122 \times 10^8$ Pa

T = Temperature, K

T_{mc} = Cladding solidus temperature, K

Care should be exercised when using the function UTS since it is for unirradiated cladding and may be inconsistent with the option used for the cladding flow stress.

8.7.10 Cladding Flow Stress

There exist four different options available to control the cladding flow stress calculation in the function YLDCF. These are controlled through the input parameter IYLD.

IYLD = 0

The value is calculated from the correlation in the NSMH [8-17] for unirradiated, 20% cold-worked, 316 SS. For temperatures below 1200 K the following form is used. It should be noted that this correlation is for the strain rate of the tensile tests and may not be appropriate in many accident cases with high strain rates.

$$\sigma_{\text{y}} = C_0 + C_1T + C_2T^2 + C_3T^3 + C_4T^4 + C_5T^5 + C_6T^6 + C_7T^7 + C_8T^8 + C_9T^9 + C_{10}T^{10} \tag{8.7-25}$$

where

σ_{y} = Clad flow stress, ksi

T = Temperature, °F

$C_0 = 9.611825 \times 10^1$

$C_1 = -1.262505 \times 10^{-1}$

$C_2 = 1.510991 \times 10^{-3}$

$C_3 = -1.021806 \times 10^{-5}$

$C_4 = 3.796623 \times 10^{-8}$

$C_5 = -8.438888 \times 10^{-11}$

$C_6 = 1.163911 \times 10^{-13}$

$C_7 = -9.993892 \times 10^{-17}$

$C_8 = 5.177194 \times 10^{-20}$

$C_9 = -1.477323 \times 10^{-23}$

$C_{10} = 1.780710 \times 10^{-27}$

The value is then converted to Pa by multiplying by 6.894757×10^6.

For temperatures above 1200 K, a form like Eq. (8.7-24) is used for the extrapolation to the melting temperature.

$$\sigma_y = \sigma_y\,(1200)\left\{1 - \left[\frac{T - 1200}{T_{mc} - 1200}\right]^2\right\} \tag{8.7-26}$$

where

$\sigma_y\,(1200) = 7.375 \times 10^7$ Pa

T = Temperature, K

T_{mc} = Clad solidus temperature, K

IYLD = 1,2

For these values of IYLD, a flow stress model developed by DiMelfi and Kramer [8-34] is used. If IYLD = 1, the model is temperature, strain, strain rate, and burnup dependent. If IYLD = 2, a high strain rate approximation is used, removing the strain rate dependence.

The flow stress, σ, is defined by

$$\sigma = \sigma_s - (\sigma_s - \sigma_1)\exp\left(-\frac{\hat{\varepsilon}}{\varepsilon_c}\right) \tag{8.7-27}$$

where

σ_1 = Yield stress of fully annealed, unirradiated material

σ_S = Saturation flow stress approached as increases

$\hat{\varepsilon}$ = Hardness parameter

ε_c = Material parameter

the hardness parameter $\hat{\varepsilon}$ has two components; ε_p, the accumulated equivalent plastic strain, and $\hat{\varepsilon}_\phi$ the hardness due to irradiation, which is determined from

$$\hat{\varepsilon}_\phi = -\varepsilon_c \ln\left(1 - \left\{1 - \exp\left[-B\left(\phi t - \phi_o t_o\right)\right]\right\}^{1/2}\right) \tag{8.7-28}$$

where

$B = 3.5 \times 10^{-27}$, m^2/n

$\phi_o t_o = 4.5 \times 10^{25}$, n/m^2

ϕt = Neutron fluence, n/m^2

and

$$\dot{\varepsilon} = \varepsilon_p + \hat{\varepsilon}_\phi + \varepsilon_{cw} \tag{8.7-29}$$

where

ε_{cw} = As-fabricated cold-work strain

The relationship between σ_s, σ_1, and ε_c is given by

$$\left[\frac{\sigma_s}{G} - \frac{\sigma_1}{G}\right]\frac{1}{\varepsilon_c} = \frac{\theta_1}{G} \tag{8.7-30}$$

where

$\theta_1/G = 3.66 \times 10^{-2}$

$G = 92.0 - 4.02 \times 10^{-2}$ T, GPa

This relationship is equivalent to assuming that the initial work-hardening rate θ_1, for annealed material ($\hat{\varepsilon} = 0$) is constant.

The functions σ_s and σ_1 are chosen to be of the form

$$\frac{\sigma_s}{G} = \frac{\sigma_{SO}}{G}\left\{1 - \exp\left[-\left(\frac{\dot{\varepsilon}_p}{\dot{\varepsilon}_{OS}}\right)^{nk}\right]\right\}^{1/k} \tag{8.7-31}$$

$$\frac{\sigma_1}{G} = \frac{\sigma_{10}}{G}\left\{1 - \exp\left[-\left(\frac{\dot{\varepsilon}_p}{\dot{\varepsilon}_{01}}\right)^{nk}\right]\right\}^{1/k} \tag{8.7-32}$$

where

$\hat{\varepsilon}_p$ = Equivalent plastic strain rate

n = Constant = 1/5.35

k = Constant = 2.0

The functions σ_{SO}, σ_{10}, $\dot{\varepsilon}_{OS}$, $\dot{\varepsilon}_{01}$ are all temperature dependent. The constant k is a nonphysical parameter that governs the sharpness of the transition between strain-rate-dependent and strain-rate-independent behavior.

If IYLD = 2, then the above relationships for σ_s and σ_1 are assumed to have the form

$$\frac{\sigma_s}{G} = \frac{\sigma_{SO}}{G} \tag{8.7-33}$$

and

$$\frac{\sigma_1}{G} = \frac{\sigma_{10}}{G} \tag{8.7-34}$$

In both cases, IYLD = 1 or IYLD = 2, the following functions are used

$$\frac{\sigma_{SO}}{G} = 2.06 \times 10^{-3} + \frac{7.12 \times 10^{-1}}{T} \tag{8.7-35}$$

$$\frac{\sigma_{10}}{G} = 2.00 \times 10^{-2} - 9.12 \times 10^{-6}T \tag{8.7-36}$$

The functions $\hat{\varepsilon}_{OS}$ and $\hat{\varepsilon}_{01}$ are strong functions of temperature and are related to the dominant creep mechanism. This is assumed to be thermally activated, so Arrhenius relationships are used.

$$\dot{\varepsilon}_{OS} = \dot{\varepsilon}_{OOS}\exp\left(-\frac{Q}{RT}\right) \tag{8.7-37}$$

$$\dot{\varepsilon}_{01} = \dot{\varepsilon}_{001}\exp\left(-\frac{Q}{RT}\right) \tag{8.7-38}$$

where

Q/R = 38,533, K

$\dot{\varepsilon}_{\text{OOS}} = 1.062 \times 10^{14}$, s^{-1}

$\dot{\varepsilon}_{001} = 3.794 \times 10^{12}$, s^{-1}

Given the above constants and relationships, it is then possible to calculate the flow stress of the cladding.

IYLD = 3

With this selection, the input flow stress table YLDTAB with its corresponding temperature table YLDTEM are searched to find the flow stress as a function of temperature. One table exists for each cladding type.

8.7.11 Thermal Conductivity of Helium

The thermal conductivity of the helium in the fuel pin free volume is temperature dependent and has been fit to the following equation [8-35].

$$K = 1.43 \times 10^{-1} + 3.17 \times 10^{-4} (T - 273) - 2.24 \times 10^{-8} (T - 273)^2 \tag{8.7-39}$$

where

K = Thermal conductivity, W m^{-1} K^{-1}

T = Gas temperature, K

This is used in the function routine HGAP.

8.7.12 Thermal Conductivity of Fission Gas

All fission gas is assumed to be xenon. A functional form of the xenon conductivity [8-35] is used in the function routine HGAP.

$$K = 5.15 \times 10^{-3} + 1.69 \times 10^{-5} (T - 273) - 3.50 \times 10^{-9} (T - 273)^2 \tag{8.7-40}$$

where

K = Thermal conductivity, W m^{-1} K^{-1}

T = Gas temperature, K

8.7.13 Fuel Hardness

In DEFORM-4 it is assumed that conditions could arise where either the fuel or cladding could be the softer material when considering fuel-cladding contact in the determination of the gap conductance. The following equation is therefore used to determine the oxide fuel hardness [8-31].

$$H = 6.009 \times 10^9 \exp\left(-\frac{T}{641}\right) \tag{8.7-41}$$

where

H = Meyer hardness, Pa

T = Temperature, K

8.7.14 Cladding Hardness

The cladding hardness is calculated from the following equation [8-21] when comparing with the fuel hardness. The softer of the fuel or cladding is used in the solid-to-solid gap conductance considerations.

$$H = 5.961 \times 10^{9} T^{0.206} , \quad T < 893.92 \text{ K} \tag{8.7-42}$$

$$H = 2.75 \times 10^{22} T^{6.53} , \quad T > 893.92 \text{ K} \tag{8.7-43}$$

where

H = Meyer hardness, Pa

T = Temperature, K

8.7.15 Cladding Fast Creep

Kramer and DiMelfi [8-34] have shown that the following equation is suitable for plastic strain of cladding at high temperatures.

$$\dot{\varepsilon} = \frac{\dot{\varepsilon}_{\text{OOS}}}{\sigma_{\text{SO}}^{n}} \sigma^{n} \exp\left(-\frac{Q_{\text{c}}}{T}\right) \tag{8.7-44}$$

where

$\dot{\varepsilon}$ = Plastic strain rate, s^{-1}

σ = Equivalent stress in the cladding, Pa

T = Temperature, K

$\dot{\varepsilon}_{\text{OOS}} = 1.062 \times 10^{14}$, s^{-1}

σ_{so} = Material constant defined in Eq. (8.7-35)

n = Stress exponent = 5.35

Q_{c} = Creep activation constant = 38,533 K

This allows for plastic deformation of the cladding below the flow stress.

8.7.16 Material Properties of Metal Alloy Fuel

Material properties for the metal fuels under consideration for the Integral Fast Reactor concept are currently under investigation in order to prepare a set of recommended values for use by the safety community. Unlike the oxide fuel which has been under study for a number of years, the metal fuel data is fairly sparse and limited to specific fuel types. Therefore, the correlations given below are to be considered preliminary in nature, but they should provide results that are consistent with future results. These have been coded into DEFORM in a manner that makes future modifications a simple replacement task.

Several thermal properties are also used by DEFORM and these have been addressed in Chapter 10 of this document. Most of the properties given below are obtained from a survey of data carried out for the modification of the FPIN code to analyze metal fuels [8-36].

8.7.16.1 Modulus of Elasticity

Although the modulus of elasticity varies with temperature, composition, and phase, a single average value has been used.

$$E_{\mathrm{f}} = 1.4 \times 10^{12} \tag{8.7-45}$$

where

E_{f} = Modulus of elasticity, Pa

8.7.16.2 Poisson's Ratio

Poisson's ratio is an input parameter to the SAS4A code. For both the ternary and fission alloy fuels the currently recommended value is 0.23.

8.7.16.3 Fracture Strength

Unlike the oxide fuels, the metal fuel does not exhibit a fracture phenomenon. While there does appear to be a separation between the phases that result in the ternary fuels, there is no radial cracking. This appears to be the result of the softness of the metal matrix and the inherent structural differences that exist between ceramics and metals. When metal fuel has been specified, the function SIGFRA returns a large value for the fracture strength to avoid cracking during the mechanical calculations.

8.7.16.4 Fuel Creep Rate

One of the more important correlations used by DEFORM is the creep rate of the fuel because of its influence on the rate of fuel swelling through the process of fission gas bubble volume change. For the metal fuels the creep rate, determined in a study performed by John Kramer [8-37], has been employed for both types of metal fuel currently considered in DEFORM.

For uranium phases existing below 973 K, the rate is dependent on two stress related terms.

$$R_{\mathrm{c}} = \left(0.5 \times 10^{4}\sigma + 6.0\sigma^{4.5}\right) \exp\left(-\frac{26168}{T}\right) \tag{8.7-46}$$

where

R_{c} = Fuel creep rate, s^{-1}

Σ = Stress, MPa

T = Temperature, K

For the high temperature gamma uranium phase, the rate is give by a single stress dependent function.

$$R_{\mathrm{c}} = 8.0 \times 10^{-2}\sigma^{3} \exp\left(-\frac{14353}{T}\right) \tag{8.7-47}$$

The time constant for bubble expansion is the inverse of the calculated creep rate. These functions are coded into the subroutine FSWELL.

8.7.17 15-15Ti Cladding Properties

8.7.17.1 Thermal Creep

Given the cladding equivalent stress and clad mid-wall temperature at each axial location, the thermal creep correlation computes the thermal creep strain rate at each time step during normal operation as well as transients. The thermal creep equation derived in [8.39] for 15-15Ti cladding has been adopted as is. The form of the equation is given as follows:

$$\dot{\epsilon}_{th,\ Eq} = [sinh(\beta\sigma_{\text{Eq}})]^n\, e^{-\frac{E}{\text{RT}}} \tag{8.7-48}$$

$$E = E_1 + E_2 T + E_3 T^2$$

Where $\dot{\epsilon}_{th,Eq}$ is the thermal creep strain rate (1/s), σ_{Eq} is the equivalent stress (MPa), and T is the clad mid-wall temperature (K). R is the gas constant, 8.3144621 J/mol/K, and E is the activation energy (J/mol). β, n, and E_1, E_2, and E_3 are the fitting parameters, which are given in Table 8.7.1 below.

Table 8.7.1: Fitting parameters for the 15-15Ti Thermal Creep Model

Fitting Parameter	Value
β (1/MPa)	1.4994E-2
E_1 (J/mol)	-6.591E+5
E_2 (J/mol)	1.8211E+3
E_3 (J/mol)	-1.0513
n	2.3319

8.7.17.2 Irradiation Creep

Given the cladding hoop stress, clad mid-wall temperature, and clad dose rate, which is a function of clad fast neutron flux, at each axial location, the irradiation creep correlation computes the irradiation creep strain rate at each time step. The irradiation creep equation derived in [8.40] for 15-15Ti cladding has been adopted as is. The form of the equation is given as follows:

$$\dot{\epsilon}_{ir,\ \theta} = 0.75 \times A \times \sigma_\theta \times \dot{\varphi}_d \tag{8.7-49}$$

Where $\dot{\epsilon}_{ir,\ \theta}$ is the irradiation creep strain rate (1/s), σ_θ is the clad hoop stress (MPa), and $\dot{\varphi}_d$ is the clad dose rate (dpa/s). The cladding dose rate is computed by multiplying the fast neutron flux with the dose conversion factor, **CDOSECONV**. The fast neutron flux is computed by multiplying **FLTPOW** with the linear heat rate. A is the fitting parameter given as a function of clad mid-wall temperature, which is shown in Figure 8.7.1. Below 673 K and above 898 K, the value of A is set to its value at 673 K and 898 K, respectively.

8.7.17.3 Void Swelling

Void swelling rate of cladding is modeled as a function of cladding dose and cladding dose rate, which is given as a function of fast neutron flux and temperature. The cladding dose rate is computed by multiplying the fast neutron flux with the dose conversion factor, **CDOSECONV**. The fast neutron flux is computed by multiplying **FLTPOW** with the linear heat rate. The void swelling plots given in Figure 8.7.2 [8.40] as a function of cladding dose and clad temperature are used to derive linear piecewise continuous functions to predict the void swelling rate at a given time step.

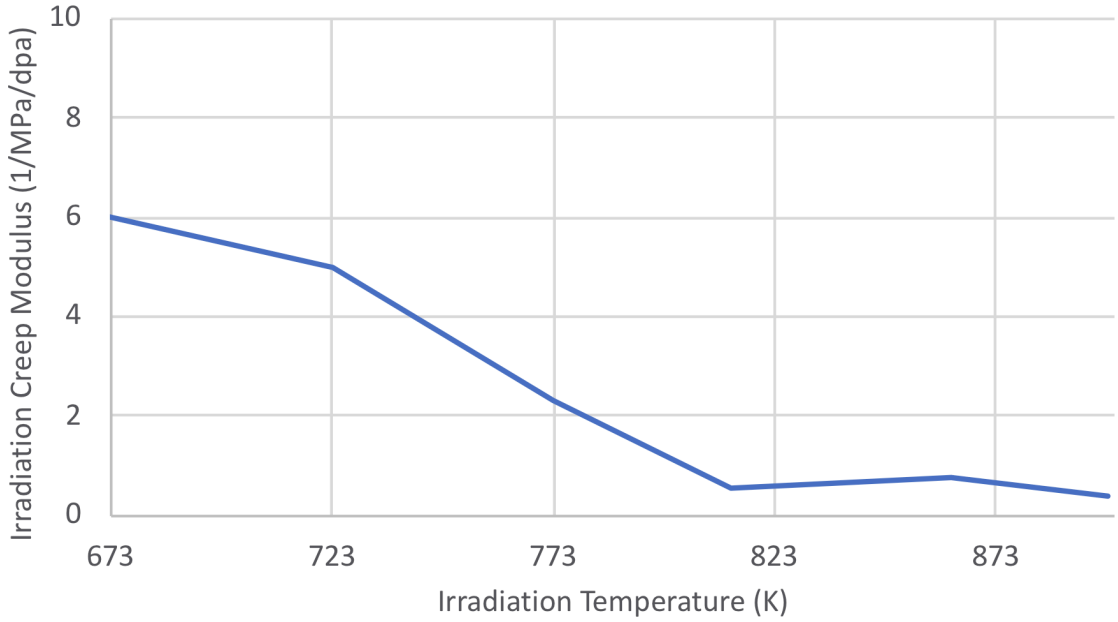


Figure 8.7.1: Irradiation Creep Modulus for 15-15Ti Cladding

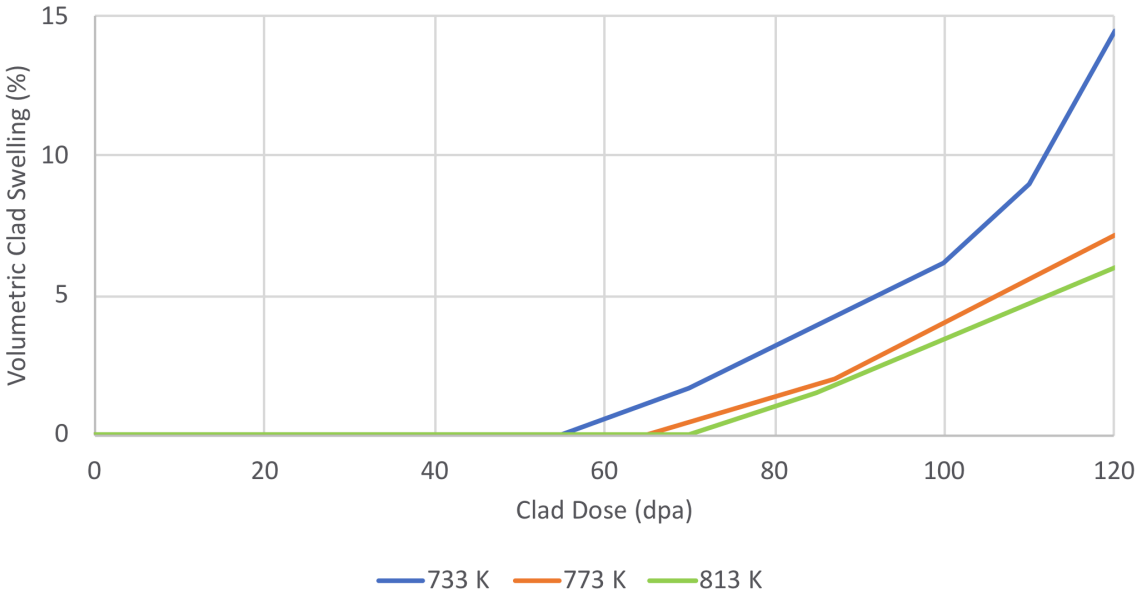


Figure 8.7.2: Experimental Data for volumetric clad swelling as a function of dose at 733 K, 773 K and 813 K for 15-15Ti cladding

8.7.17.4 Transient Creep Rupture

The creep rupture margin of the cladding is computed using the CDF/life fraction model and is a function of hoop stress, mid-wall temperature and dose of the cladding. CDF at any time, t_0, during the transient is described by the following relation:

$$\text{CDF}\,(t_0) = \int_0^{t_0} \frac{\text{dt}}{t_r(\sigma_{\text{eq}}, T, \varphi_d)} \tag{8.7-50}$$

where, t_r is the time to rupture (s), σ_{eq} is the equivalent stress (MPa), T is the temperature (K), and φ_d is the dose (dpa).

Time to rupture correlations given in [8.39] for the unirradiated 15-15Ti have been extended in order to account for the neutron irradiation induced damage and adopted in this model. The following form is used:

$$P = \begin{cases} \left[\frac{A_1}{A_3}\left(\log\sigma_{\text{eq}} - A_2\right) + 1\right]\left[1 + A_3 - e^{A_4(\log\sigma_{\text{eq}} - A_2)}\right] & T < 1023.15 \\ B_1 \log\sigma_{\text{eq}} + B_2 & T \geq 1023.15 \end{cases} \tag{8.7-51}$$

$$t_r = i_f \times 10^{\left(P\frac{1000}{T} - 17.6\right)} \times 3600$$

where, P is the Larson-Miller parameter and A_1, A_2, A_3, A_4, B_1, and B_2 are the fitting parameters given in Table 8.7.2.

Table 8.7.2: Fitting parameters for 15-15Ti Time to rupture correlation

Fitting Parameter	Value
A_1	-0.78
A_2	2.3054
A_3	19.512
A_4	5.196
B_1	-6.1175
B_2	31.7882

The modeling of the irradiation induced embrittlement of 15-15Ti was approximated using clad void swelling data and the interrelation between void swelling and clad embrittlement for similar stainless steels such as SS316 and D9 using the creep rupture data given in Refs. [8.41], [8.42], and [8.43]. The i_f correction factor is modeled as a function of 15-15-Ti cladding dose as given in Figure 8.7.3. The cladding dose rate is computed by multiplying the fast neutron flux with the dose conversion factor, **CDOSECONV**. The fast neutron flux is computed by multiplying **FLTPOW** with the linear heat rate. At high doses where void swelling takes off exceeds 5% and embrittles cladding, the irradiation factor is assumed to be 0.01 consistent with the findings for SS316 cladding. This value is also used for cladding doses that are beyond 120 dpa.

In order to model the stochastic nature of clad failure, a normal probability density function (PDF) has been adopted as a function of the logarithm of the CDF.

$$\text{PDF} = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left[-\frac{\left(\log\text{CDF} - \mu\right)^2}{2\sigma^2}\right] \tag{8.7-52}$$

where σ is the standard deviation and μ is the mean in units of the logarithm of CDF.

The integral of the PDF given by the equation above is interpreted as the fraction of the fuel pins that are failed. PDF and integral of PDF for $\sigma = 1$ and $\mu = 0$ are plotted in Figure 8.7.4. The model currently ignores potential pin-to-pin failure propagation due to fission gas and fuel ejection to the channel, clad ballooning, and blockages.

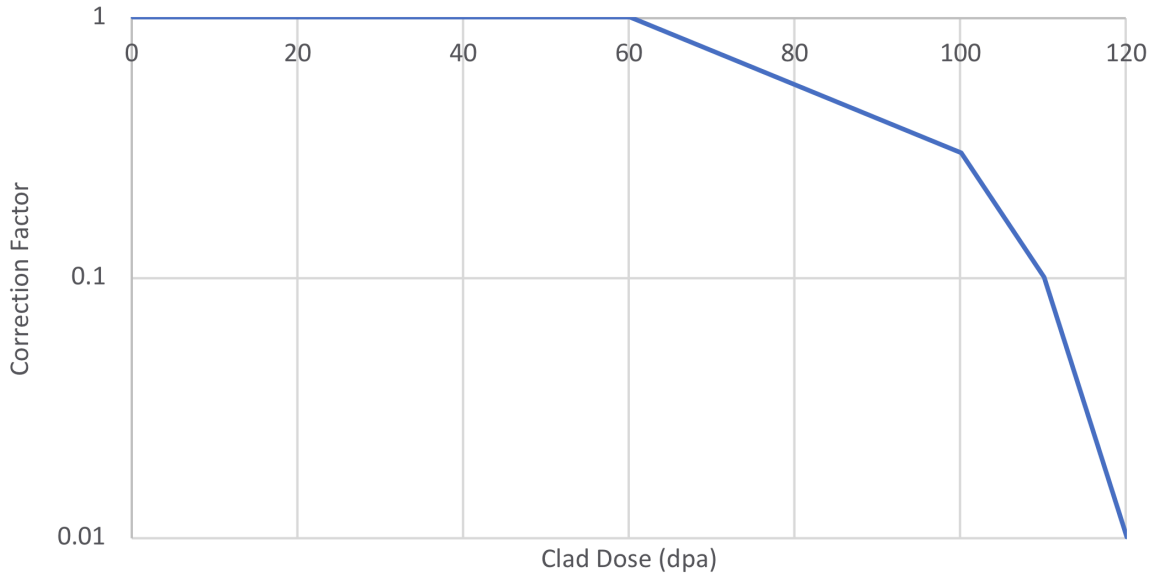


Figure 8.7.3: Approximate irradiation correction factor for 15-15Ti time to rupture as a function of cladding dose

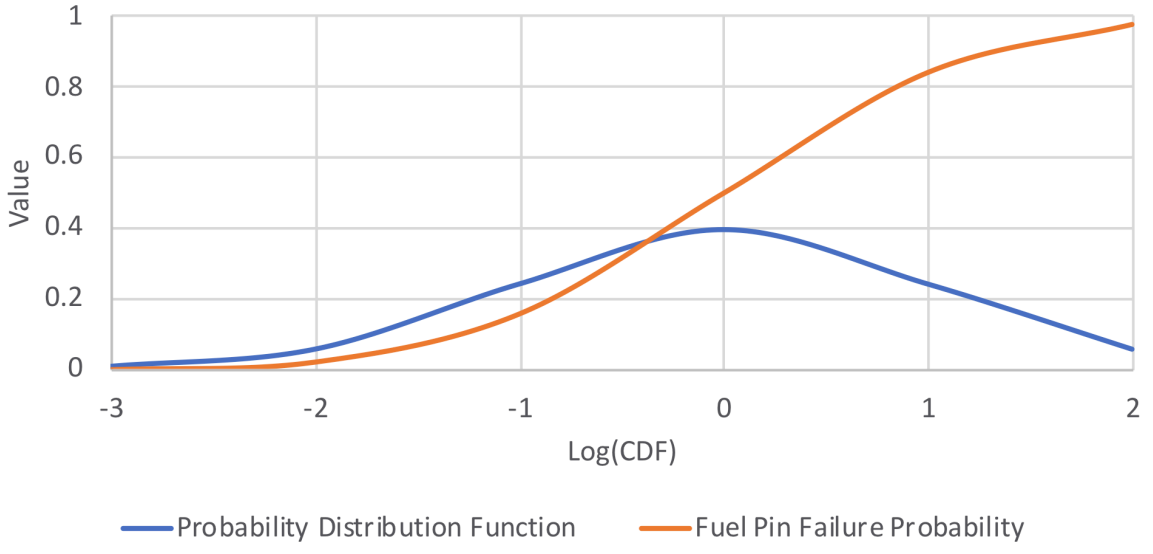


Figure 8.7.4: An example PDF and integral of PDF for $\sigma = 1$ and $\mu = 0$

8.7.17.5 Cladding Outer Corrosion

Chemical interactions between stainless steel clad and lead coolant is a critical issue of LFR designs. The highly complex nature of the problem and unavailability of corrosion data led to the decision to keep this model simple and mostly dependent on user input. The user is allowed to define the thickness of the cladding wastage prior to the transient. The corrosive layer is increased from zero to the set value during the pre-transient characterization and is kept as constant during the transient. The corroded outer layer of the cladding is assumed not to bear any applied load.

8.8 References

8-1. D.R. MacFarlane, Ed., "SAS1A, A Computer Code for the Analysis of Fast Reactor Power and Flow Transients," ANL-7606, Argonne National Laboratory, October 1970.

8-2. F.E. Dunn, et al., "The SAS2A LMFBR Accident Analysis Computer Code," ANL-8138, Argonne National Laboratory, October 1974.

8-3. F.E. Dunn, et al, Unpublished information, Argonne National Laboratory, 1975.

8-4. J.E. Cahalan and D.R. Ferguson, Ed., Unpublished information, Argonne National Laboratory, 1977.

8-5. Energy Research and Development Administration, Unpublished information, 1977.

8-6. M. Bober and G. Schumacher, "Material Transport in the Temperature Gradient of Fast Reactor Fuels," *Adv. Nucl. Sci. Tech.*, vol. 7, p. 121-196, 1973.

8-7. C.F. Clement, "The Movement of Lenticular Pores in UO_2 Nuclear Fuel Elements," *J. Nucl. Mat.*, vol. 68, p. 63-68, 1977.

8-8. W.F. Lackey, F.J. Homan, A.R. Olsen, "Porosity and Actinide Redistribution During Irradiation of (U Pu) O_2," *Nucl. Technol.*, vol. 16, p. 120-142, 1972.

8-9. F.A. Nichols, "The Theory of Grain Growth in Porous Compacts," *Journal of Applied Physics*, vol. 37, no. 13, p. 4599-4602, 1966.

8-10. R.N. Singh, "Isothermal Grain Growth Kinetics in Sintered Pellets," *J. Nucl. Mat.*, vol. 64, p. 174-178, 1977.

8-11. G.A. Reymann, Ed., "MATPRO-VERSION 10: A Handbook of Materials Properties for Use in the Analysis of Light Water Reactor Fuel Rod Behavior," TREE-NUREG-1180, EG & G Idaho, Inc., 1978.

8-12. J.B. Ainscough, B.W. Oldfield and J.O. Ware, "Isothermal Grain Growth Kinetics in Sintered UO_2 Pellets," *J. Nucl. Mat.*, vol. 49, p. 117-128, 1973-74.

8-13. J. Rest, Unpublished information, Argonne National Laboratory, 1978.

8-14. E.E. Gruber, "A Generalized Parametric Model for Transient Gas Release and Swelling in Oxide Fuels," *Nucl. Technol.*, vol. 35, p. 617-634, October 1977.

8-15. J. Weisman, et al., "Fission Gas Release from UO_2 Fuel Rods with Time-Varying Power Histories," *Trans. Am. Nucl. Soc.*, vol. 12, p. 900, 1969.

8-16. J.A. Dearien, et. al. "FRAP-S2: A Computer Code for the Steady State Analysis of Oxide Fuel Rods," TREE-NUREG-1107, EG & G Idaho, Inc., July 1977.

8-17. Hanford Engineering Development Laboratory, Unpublished information, 1971.

8-18. A.M. Ross, R.L. Stoute, "Heat Transfer Coefficients Between UO_2 and Zircalloy 2," AECL-1552, Atomic Energy of Canada, LTD, 1962.

8-19. G.R. Horn and F.E. Panisko, Unpublished information, Hanford Engineering Development Laboratory, 1972.

8-20. M. Knudsen, "The Kinetic Theory of Gases–Some Modern Aspects," Methuen and Co, Ltd, 2nd Edition, 1946.

8-21. Lassmann and F. Pazdera, "URGAP, A Gap Conductance Model for Transient Conditions," Water Reactor Fuel Element Performance Modeling, edited by J. Gittos, Applied Science Publishers, Chapter 3, pp. 97-113, 1983.

8-22. W.R. Bohl, et al., Unpublished information, Argonne National Laboratory, 1975.

8-23. C. Essig, et al., "Dynamic Behavior of RAPSODIE in Exceptional Transient Experiments, "CONF-850410, Proc. of the Intl. Topical Meeting on Fast Reactor Safety, Knoxville, Tennessee, p. 635, April 1985.

8-24. E. E. Gruber and J. M. Kramer, Unpublished information, Argonne National Laboratory, 1985.

8-25. L. C. Walter, B. R. Seidel, and J. H. Kittel, "Performance of Metallic Fuels and Blankets in Liquid-Metal Fast Breeder Reactors," *Nucl. Technol.*, vol. 65, p. 179, 1984.

8-26. B. R. Seidel and L. C. Walters, "EBR-II Metallic Driver Fuel - A Live Option," *J. Eng. Power*, vol. 103, p. 612, 1981.

8-27. G. Birgersson, et al., Unpublished information, Argonne National Laboratory, 1983.

8-28. K. J. Miles, Jr. and Kalimullah, "The Inherent Safety Phenomenon of Fission-Gas Induced Axial Extrusion in Oxide and Metal Fueled LMFBRs," CONF-850410, Proc. of the Intl. Topical Meeting on Fast Reactor Safety, Knoxville, Tennessee, p. 103, April 1985.

8-29. W. R. Robinson, et al., Unpublished information, Argonne National Laboratory, 1985.

8-30. F.E. Bard, Unpublished information, Hanford Engineering Development Laboratory, 1982.

8-31. D.S. Dutt, et al., Unpublished information, Hanford Engineering Development Laboratory, 1973.

8-32. S.Y. Ogawa, E.A. Lees and M.F. Lyons, "Power Reactor High Performance UO_2 Program, Fuel Design Summary and Progress Status," GEAP-5591, General Electric Company, 1968.

8-33. P. Verbeek and H. Hoppe, "COMETHE-IIIJ: A Computer Code for Predicting Mechanical and Thermal Behavior of a Fuel Pin, Part 1, General Description," INBFR-107 (original report BN-7609-07, Belgonucleaire, Brussels, Belgium, 1977.

8-34. R.J. DiMelfi and J.M. Kramer, "Modeling the Effects of Fast-Neutron Irradiation on the Subsequent Mechanical Behavior of Type 316 Stainless Steel," *J. of Nucl. Mat.*, vol. 89, no. 2 & 3, pp. 338-346, 1980.

8-35. R.B. Baker, "Calibration of a Fuel-to-Cladding Gap Conductance Model for Fast Reactor Fuel Pins," HEDL-TME-77-86, Hanford Engineering Development Laboratory, May 1978.

8-36. T. H. Hughes, Unpublished information, Argonne National Laboratory, 1985.

8-37. J. M. Kramer, private communication, October 1984.

8-38. D.R. Olander, "Fundamental Aspects of Nuclear Reactor Fuel Elements," TID-26711-Pl, Energy Research and Development Administration, 1976.

8.39. N. Cautaerts, et al., "Thermal Creep Properties of Ti-Stabilized DIN 1.4970 (15-15Ti) Austenitic Stainless Steel Pressurized Cladding Tubes," *Journal of Nuclear Materials*, Vol. 493, pp. 154-67, (2017).

8.40. J. L. Seran, et al., "Behavior under Neutron Irradiation of the 15-15Ti and EM10 Steels Used as Standard Materials of Phenix Fuel Assemblies," ASTM STP 1125, pp. 1209-1233, (1992).

8.41. N. Miyaji, et al., "Post-irradiation Creep Rupture Properties of FBR grade 316 SS Structural Material," *Journal of Nuclear Materials*, Vol. 271, pp. 173-178, (1999).

8.42. S. Ukai, et al., "In-reactor Creep Rupture Properties of 20% CW Modified 316 Stainless Steel," *Journal of Nuclear Materials*, Vol. 278, pp. 320-327, (2000).

8.43. R. J. Puigh, et al., "In-Reactor Creep Rupture Behavior of the D19 and 316 Alloys," *Influence of Radiation on Material Properties: 13th International Symposium – Part II*, ASTM STP 956, pp. 22-29, (1987).

8.9 Nomenclature

Symbols	Definition	Units
A, A_{h}	Axial segment length	m
B	Burnup	atom %
D	Grain size	m
C_{r}	Calibration constant for metal fuel bubble radius	
E	Modulus of elasticity	Pa
E_{c}	Energy of the cladding	J/kg
E_{f}	Modulus of Elasticity of the fuel	Pa
F_{as}	Fraction of volume change occurring axially	
F_{f}	Failure fraction	
f	Mole fraction	
G_{f}	Fission gas retained in fuel cell	kg
G_{r}	Fission gas released from fuel cell	kg
G_{t}	Total fission gas produced in fuel cell	kg
g	Thermal jump distance	m
F	Force	N
H	Hardness of cladding	Pa
H_{sp}	Height of sodium in the plenum	m
h	Heat-transfer coefficient	W $m^{-2}K^{-1}$
K	Bulk modulus	Pa
k	Thermal conductivity	W $m^{-1}K^{-1}$
M	Molecular weight, g mole, or Mass	kg
$M_{\mathrm{s,j}}$	Mass of sodium in fuel-cladding gap in axial segment j	kg
M_{s1}	Mass of sodium loaded into the fuel pin	kg
M_{sp}	Mass of sodium in the plenum region	kg
M_{st}	Total mass of sodium in the fuel-cladding gap	kg
n	Number of moles	
P	Pressure	Pa
P_{cav}	Molten cavity pressure	Pa
P_{ext}	Coolant channel pressure	Pa
P_{fci}	Fuel-Cladding interface pressure	Pa
p	Porosity fraction	
R	Universal gas constant	J K^{-1}g-$mole^{-1}$
R_{B}	Fission gas bubble radius	m
R_{c}	Fuel creep rate	s^{-1}
R_{f}	Fission gas release fraction	
r	Radius	m
r_{cav}	Radius of central molten cavity	m
r_{ci}	Inner radius of the cladding	m
r_{co}	Outer cladding radius	m
r_{cl}	Outer boundary of SAS4A inner cladding cell	m

continues on next page

Table 8.9.1 – continued from previous page

Symbols	Definition	Units
r_{c2}	Outer boundary of SAS4A central cladding cell	m
r_{fo}	Outer fuel radius	m
r_p	Inner cladding radius in the plenum	m
r_{ci}	Volumetric swelling fraction	
r_{co}	Temperature	K
r_{ci}	Average temperature	K
r_{cz}	Temperature of cladding center	K
r_{fo}	Inner cladding temperature	K
r_p	Outer cladding temperature	K
s_f	Temperature of PLUTO2/LEVITATE inner cladding node	K
T	Temperature	K
T_a	Average temperature	K
T_{cc}	Temperature of cladding center	K
T_{ci}	Inner cladding temperature	K
T_{co}	Outer cladding temperature	K
T_{cl}	Temperature of PLUTO2/LEVITATE inner cladding node	K
T_{eut}	Eutectic temperature	°C
T_f	Time to failure	s
t	Time	s
U	Velocity	m s^{-1}
U_{ts}	Ultimate tensile strength	Pa
u	Radial displacement	m
V	Volume	m^3
V_c	Volume of a cell	m^3
V_f	Volume of fission gas in closed porosity	m^3
V_r	Volume of fission gas moved from closed to open porosity	m^3
ΔV_a	Axial volume change	m^3
ΔV_{exc}	Volume of excess radial fuel swelling	m^3
ΔV_r	Radial volume change	m^3
V_{th}	Theoretical volume of material in a cell	m^3
v	Circumferential displacement	m
W	Weighting factor	
z_o	Radially constant axial strain	
Z	Axial elevation	m
α	Thermal-expansion coefficient	K^{-1}
γ	Surface tension	N m^{-1}
Δt	Time-step length	s
δ	Mean surface roughness	m
ε	Strain or emissivity	
η	Outer radius of solid fuel annulus	m
ρ	Inner radius of solid fuel annulus or density	m kg m^{-3}
$\rho s, j$	Density of sodium in fuel-cladding gap in axial segment j	kg m^{-3}
ρsp	Density of sodium in the plenum	kg m^{-3}
υ	Poisson's ratio	
ϕ	Neutron flux	n $m^{-2}s^{-1}$
σ	Stress, Pa, or Stefan-Boltzmann constant	J $s^{-1}m^{-2}k^{-4}$
Y	Time constant, s^{-1}, or incubation parameter	n/m^2
Σ	Negative fractional density change in the cladding	

Superscripts	
f	Force component
th	Thermal component
p	Plenum
v	Central fuel void

Subscripts	
r	Radial or reference
θ	Circumferential
z	Axial
f	Force or fuel
th	Thermal
ρ	Inner radius of the solid fuel annulus
η	Outer radius of the solid fuel annulus
g	Plenum gas or fuel-cladding gap
FC	Fuel-cladding interface
c	Cladding
m	Maximum
k	Iteration counter
i	Time step or radial node
j	Axial node
B	Bubble
fg	Fission gas
s	Swelling
v	Central fuel void
p	Plenum or pore
He	Helium
T	Total
R	Reference
cav	Molten cavity
mc	Melting of cladding

CHAPTER

NINE

MFUEL: METALLIC FUEL PERFORMANCE MODEL

9.1 Introduction

In Version 5.7, a new metallic fuel performance model for pre-transient and transient analysis has been introduced. The new fuel model is described in the remaining sections of this chapter, the description of DEFORM-5, which is considered deprecated, has been moved to Section 9.8.8.

The integrated metallic fuel performance model accounts for the coupled nature of the uranium plutonium zirconium metallic fuel phenomena and its impact/evolution on/during safety analysis. Given the user-provided operating history, initial geometry, fresh fuel composition, mass, cladding type, and the calculated fuel pin and coolant temperature distribution, MFUEL, the metallic fuel performance model, simulates the fuel and cladding behavior as a function of time. The main capabilities of MFUEL are prediction of: 1) Fuel pin mechanics, and compositional and dimensional changes; 2) Clad failure; and 3) Fuel pin thermal resistance. Achieving these high-level goals accurately is strongly related to the model performance of individual physical processes taking place within the fuel pin during operation. Metal fuels typically operate above half of their solidus temperatures during normal as well as off-normal conditions. Availability of thermal activation provides the driving force for various diffusional processes leading to complex phase transformations, micro-structure evolution, significant amount of fuel swelling, interconnected porosity formation, excessive amounts of fission gas release, and fuel clad chemical interactions. Clad failure in fast reactors primarily occurs as a result of creep rupture augmented by clad wastage formation. The reaction is driven by thermal creep induced dislocation motion, grain boundary cavity nucleation, and growth and breakup of grain boundaries. The high-level complexity and limited available data requires introducing physics-based modeling approaches to gain extrapolation ability and sensitivity with respect to various conditions.

As shown in Figure 9.1.1, the MFUEL driver evaluates the different physical phenomena, described in Section 9.2 at each time step. At the end of each time step, the MFUEL driver updates the fuel pin geometry, fuel pin composition, plenum pressure, fuel porosity, bond sodium infiltration, and clad failure margin. Each of these quantities has an impact on the fuel pin thermal-hydraulics, reactivity feedback, and margin to fuel melting.

9.2 Model Formulation

9.2.1 Fuel Constituent Redistribution

Fuel constituent redistribution is one of the important characteristics of metallic fuel. Under a typical temperature gradient, three distinct radial regions form in the fuel [9-1].

- A body centered cubic (BCC) single γ phase region forms in the central radial region.

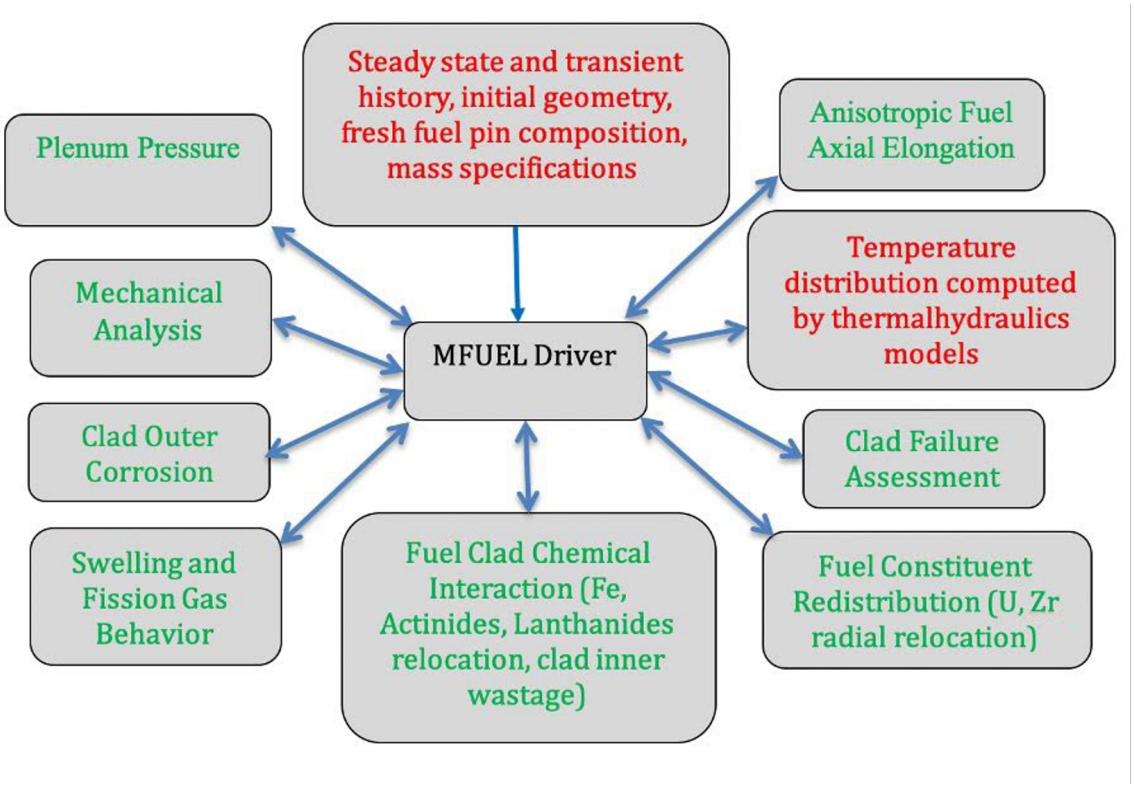


Figure 9.1.1: MFUEL - Integrated Metal Fuel Models. Processes shown in red are handled by other SAS models and processes shown in green are handled by the MFUEL driver.

- An effective dual phase structure forms in the middle radial region. This can approximately be characterized by tetragonal β precipitates and BCC γ phase regions.
- In the outer radial region, an approximate dual phase region forms, characterized by orthorhombic α precipitates and BCC δ phase regions.

Due to the difference of chemical affinity between the formed phase regions and zirconium and uranium atoms, radial migration of uranium and zirconium takes place. At equilibrium, zirconium-rich, uranium-depleted phases form in the central region, zirconium-depleted, uranium-rich phases form in the middle radial region, and a slightly zirconium-enriched region forms in the outer region. These structural and compositional changes affect fission gas behavior, porosity evaluation, fuel sintering behavior, solidus and liquidus temperature, fuel thermal conductivity, and radial power distribution.

A one-dimensional diffusion model based on thermo-transport theory is adopted to predict the fuel constituent redistribution. Based on experimental observations, it is assumed that: U and Zr atoms switch places, Pu is immobile, enthalpy of solution is negligible, and kinetic reactions terminate when a solubility limit is reached. Eq. (9.2-1) is used as the continuity equation for the Zr migration in metallic alloy fuel.

$$\frac{du}{dt} = -\nabla \bullet (J_D + J_T) \tag{9.2-1}$$

where the diffusive transport in single-phase regions is given as:

$$J_D = -D\nabla u \tag{9.2-2}$$

and the thermal transport term is given as follows:

$$J_T = -\nabla T \frac{DQ}{RT^2} u \tag{9.2-3}$$

A summation over local phases is implied for thermal transport.

Axial migration is assumed to be negligible. In one-dimensional coordinates, Eq. (9.2-1) becomes:

$$\frac{du}{dt} = \nabla \bullet \left[\left(D\nabla + \nabla T \frac{DQ}{RT^2} \right) u \right] = \frac{1}{r} \frac{d}{dr} \left(rD \frac{du}{dr} + r \frac{dT}{dr} \frac{DQ}{RT^2} u \right) \tag{9.2-4}$$

where u is the Zr atom fraction, D is the Zr diffusion coefficient (m^2/s), Q is the heat of transport of a local phase (J/mol), T is the temperature (K), R is the universal gas constant (J/mol/K), and r is the radial coordinate (m).

9.2.1.1 Numerical Solution

Utilizing a finite volume approach and applying the divergence theorem to Eq. (9.2-4), the following discretized equation for Zr atom fraction is obtained for the single-phase region:

$$\frac{d\overline{u}_i}{dt} \frac{r_{i+\frac{1}{2}}^2 - r_{i-\frac{1}{2}}^2}{2} = \left[rD \frac{du}{dr} \right]_{r_{i-\frac{1}{2}}}^{r_{i+\frac{1}{2}}} + \left[r \frac{dT}{dr} \frac{DQ}{RT^2} u \right]_{r_{i-\frac{1}{2}}}^{r_{i+\frac{1}{2}}} \tag{9.2-5}$$

In two-phases regions, Fick's Law does not apply and the diffusion term is neglected:

$$\frac{d\overline{u}_i}{dt} \frac{r_{i+\frac{1}{2}}^2 - r_{i-\frac{1}{2}}^2}{2} = \left[r \frac{dT}{dr} \frac{DQ}{RT^2} u \right]_{r_{i-\frac{1}{2}}}^{r_{i+\frac{1}{2}}} \tag{9.2-6}$$

where i is the radial node, u is the Zr atom fraction, D is the Zr diffusion coefficient (m^2/s), Q is the heat of transport (J/mol), T is the temperature (K), R is the universal gas constant, (J/mol/K), and r is the radial dimension (m).

For initial conditions, a uniform zirconium concentration is assumed based on user-defined input.

9.2.1.2 Boundary Conditions

Assuming no Zr leaves or enters at radial boundaries, the following boundary conditions are applied to the fuel inner and outer surfaces:

$$\frac{du}{dr} = 0 \;\&\; Q = 0 \; at \; r = 0 \tag{9.2-7}$$

$$\frac{du}{dr} = 0 \;\&\; Q = 0 \; at \; r = R \tag{9.2-8}$$

where R is the fuel outer surface (m) and Q represents the heat of transport (J/mol) for each phase present at the corresponding boundary.

In the MFUEL model, Zr diffusion is bounded by the solubility limits of the phases present. To simplify the numerical solution while applying the solubility limit conditions, the following conditions are enforced:

$$\text{If: } u_i \geq u_{sol,\gamma}, u_{i+1} \geq u_{sol,(\gamma \; or \; \delta)} \text{ Then: } Q_{\beta(i,i+1)} = 0 \tag{9.2-9}$$

$$\text{If: } u_i \geq u_{sol,\gamma}, \; u_{i-1} \geq u_{sol,\gamma} \text{ Then: } Q_{\gamma(i,i-1)} = 0 \tag{9.2-10}$$

$$\text{If: } u_i < u_{sol,\beta} \text{ Then: } Q_{\gamma(i,i-1)} = 0 Q_{\beta(i,i+1)} = 0 \tag{9.2-11}$$

$$\text{If: } u_i < u_{sol,\alpha} \text{ Then: } Q_{\delta(i,i+1)} = 0 Q_{\alpha(i,i+1)} = 0 \tag{9.2-12}$$

where $u_{sol,\alpha}$, $u_{sol,\beta}$, $u_{sol,\delta}$, $u_{sol,\gamma}$ are the solubility limits of $\alpha, \beta, \delta, \gamma$ phases, respectively, $Q_{\alpha(i,i+1)}$, $Q_{\beta(i,i+1)}$, $Q_{\delta(i,i+1)}$ are the heat transport between cell i and $i+1$ for α, β, and δ phases, respectively, and $Q_{\gamma(i,i-1)}$ is the heat of transport between cell i and $i-1$ for the γ phase.

9.2.1.3 Time Step Control

In order to assure numerical convergence, the time step for the fuel constituent redistribution model is selected as a function of the fuel centerline temperature as given in Eq. (9.2-13).

$$\Delta t_{Zr} = min \left(\begin{cases} 50 \; s & T_c \geq 1050 \; K \\ 1000 \; s & 900 \leq \; T_c < 1050 \; K \\ 2000 \; s & T_c < 900 \; K \end{cases} , \Delta t_{MFUEL} \right) \tag{9.2-13}$$

where Δt_{MFUEL} is the MFUEL's main time step, Δt_{Zr} is the selected time step for the Zr migration model, and T_c is the fuel centerline temperature (Kelvin). Section 9.3 describes MFUEL's main time step for steady state and transients.

9.2.2 Fuel Clad Chemical Interaction

One of the constraints limiting metallic fuel performance is its chemical interaction with steel cladding. Its low solidus temperature combined with operation above the mid-point temperature of melting provides thermal activation for diffusion. During normal operation, migration of lanthanides from the fuel to the inner surface of the cladding leads to the formation of a brittle layer on this surface [9-2]. In addition, diffusion of iron from the cladding to the fuel and formation of low melting temperature phases at the fuel surface occurs upon contact between the fuel and cladding. These iron-bearing phases may lead to eutectic formation during transients at elevated temperatures. Eutectic formation actuates the iron dissolution/penetration and can result in a significant amount of clad wastage [9-3]. The lanthanide-rich, brittle layer and eutectic layer lead to clad thinning and embrittlement, which can result in creep fracture of the cladding.

9.2.2.1 Lanthanide Migration

Lanthanides are mostly insoluble within metallic fuels [9-4]. Hence, it is assumed that they can be treated as diffusing in a single-phase medium. Having been produced by fission, the model allows lanthanides to diffuse under a concentration gradient. Lanthanide atoms that reach the unreacted cladding interface are assumed to precipitate in the form of $(Fe, Cr)_{17}LA_2$, consistent with experimental observations [9-5].

The continuity equation for lanthanide migration in metallic alloy fuel is written as:

$$\frac{du}{dt} = -\nabla \cdot \mathbf{J}_D + q_{\text{La}} \tag{9.2-14}$$

where u is the atom density per unit volume (#/m^3), $\mathbf{J}_D$ is the diffusive transport current given in Eq. (9.2-2) , $q_{\text{La}} = Y_{La} q'''$ is the fission-produced lanthanide source term (#/m^3/s), Y_{La} is the lanthanide yield per Joule, and q''' is the power density, W/m^3.

In one-dimensional coordinates, the equation becomes as follows:

$$\frac{du}{dt} = D\nabla \cdot \nabla u + q_{\text{La}} = \frac{1}{r}\frac{d}{dr}\left(rD\frac{du}{dr}\right) + Y_{La} q''' \tag{9.2-15}$$

Initial lanthanide concentration is assumed to be zero for metal alloy fuel. Prior to the contact between the fuel and cladding, lanthanides are produced within the fuel but assumed to not be trapped in the fuel. Upon the formation of the contact between fuel and cladding, it is assumed that lanthanides diffuse to the cladding and form $(Fe, Cr)_{17}LA_2$ phase. At the interface between $(Fe, Cr)_{17}LA_2$ bearing clad layer and unaffected cladding, the boundary condition for the free lanthanide concentration is zero. The hard contact between the fuel and cladding initiates when the fuel outer surface reaches the inner clad surface where the brittle lanthanide layer presents. The brittle layer formed as a result of lanthanide diffusion is assumed not to bear any load. Hence, it is removed from the available cladding thickness during mechanical analysis.

9.2.2.2 Numerical Solution

Lanthanide diffusion is solved in the radial direction from fuel centerline to the outermost cladding cell in which lanthanides have migrated. Before contact is made between the fuel and the cladding, there is no lanthanide migration into the cladding. Once contact is made, lanthanides begin to diffuse into the inner-most cladding cell. Diffusion into subsequent (outward) cells is limited until a solubility limit is reached in the current, outermost cell. Once the outermost cell reaches saturation, diffusion can progress to the next outward cell. Using a finite volume approach and applying the divergence theorem, the following discretized equation is obtained for the lanthanide diffusion through the fuel and cladding:

$$\left(\frac{d\overline{u}_i}{dt} - q_i\right)\frac{r^2_{i+\frac{1}{2}} - r^2_{i-\frac{1}{2}}}{2} = \left[rD\frac{du}{dr}\right]^{r_{i+\frac{1}{2}}}_{r_{i-\frac{1}{2}}} \tag{9.2-16}$$

where i is the radial cell, u is the La atom density (#/m^3), q_i is the lanthanide production rate (#/m^3/s), D is the La diffusion coefficient (m^2/s), and r is the radial coordinate (m).

The production of lanthanides, q_i, can be related to the fission power, P_i, in a region of the core with Volume, V_i. If the yield of lanthanides, y_{La}, is defined per unit energy, the following equation is obtained:

$$q_i = y_{\text{La}}\frac{P_i}{V_i} \tag{9.2-17}$$

Prior to the pre-transient irradiation, MFUEL assumes there are no lanthanides in the fuel.

9.2.2.3 Boundary Conditions

$$\left.\frac{du}{dr}\right|_{r=0} = 0 \tag{9.2-18}$$

$$u\left(r = R_{\text{out}}\right) = 0 \tag{9.2-19}$$

where R_{out} is the radial boundary between lanthanide-bearing and lanthanide-free cladding cells (m). Note that lanthanides that form compounds with the cladding are not counted as free lanthanide atoms, so there is a sink at the outer boundary. Once the lanthanide concentration at boundary node reaches the solubility limit, the corresponding node is assumed not to bear load (wastage) and the boundary node is shifted to the next cladding radial node.

9.2.2.4 Time Step Control

In order to assure numerical convergence, the time step for the lanthanide migration model is selected as a function of the fuel centerline temperature as given in Eq. (9.2-20).

$$\Delta t_{La} = min\left(\begin{cases} 50\ s & T_c \geq 1050\ K \\ 200\ s & 900 \leq\ T_c < 1050\ K \\ 500\ s & T_c < 900\ K \end{cases}, \Delta t_{MFUEL}\right) \tag{9.2-20}$$

where Δt_{MFUEL} is the MFUEL main time step, Δt_{La} is the selected time step for the lanthanide migration model, and T_c is the fuel centerline temperature (Kelvin). Section 9.3 describes MFUEL's main time step for steady state and transients.

9.2.2.5 Iron Migration

Diffusion of iron into the fuel leads to formation of various complex phase structures at the fuel/clad interface. Experiments performed with U-10Zr/Fe-12Cr diffusion couples show that the main iron-bearing regions can be grouped into eight approximate regions, as given in Table 9.2.1 and depicted in Figure 9.2.1 [9-6]. The maximum iron-bearing phase forms at the fuel/clad interface. As iron penetrates towards the fuel inner regions, the equilibrium iron concentration drops in each formed phase. In this work, each region is represented with its corresponding equilibrium iron concentration. Since the phases present are complex and uphill diffusion takes place, the diffusion of iron under the chemical potential gradient is modeled using regular solution theory. Thermodynamics and kinetics parameters is approximated based on available experimental data.

Table 9.2.1: Phase names, numbering, and approximate iron solubility

Phase Number	Phases	Approximate Iron Solubility limit (%)
0	Fuel	0
1	U-Pu-Zr matrix + U-Pu-23Zr-6Fe	2.5
2	U-Pu-Zr matrix+ U-Pu-42Zr-33Fe	13.2
3	(U,Pu)6Fe + U - 42Zr - 33Fe	21.8
4	(U,Pu)6Fe + U - 32Zr - 50 Fe	28.6
5	Fe2Zr + (U,Pu)6Fe	40.5
6	(U,Pu)Fe2` i	66.7
7	Clad	0.05 (lower limit)
8	Eutectic	50

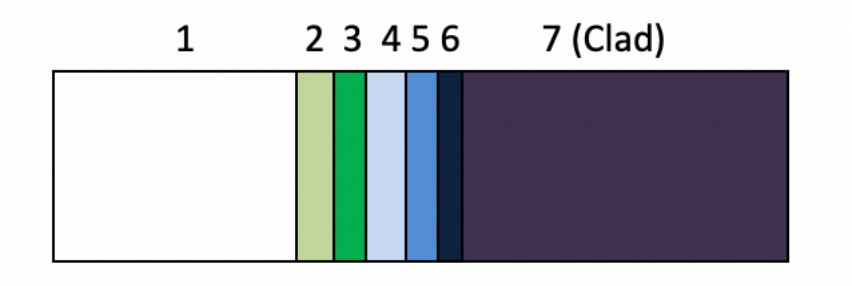


Figure 9.2.1: Phases formed at fuel cladding interface

The continuity equation for iron migration into metallic alloy fuel is defined as follows:

$$\frac{du}{dt} = -\nabla \cdot \mathbf{J}_A \tag{9.2-21}$$

where J_A, the iron diffusion current (atom fraction*m/s), is given as follows:

$$J_A = -uM\nabla\mu \tag{9.2-22}$$

In one-dimensional coordinates, Eq. (9.2-21) becomes:

$$\frac{du}{dt} = \nabla \cdot uM\nabla\mu = \frac{1}{r}\frac{d}{dr}\left(ruM\frac{d\mu}{dr}\right) \tag{9.2-23}$$

where u is the iron atom fraction, μ is the chemical potential of iron (J/mol), $M = \frac{D}{RT}$ is the mobility, and D is the diffusion coefficient (m^2/s), R is the universal gas constant (J/mol/K), and T is temperature (K).

The chemical potential is defined for each phase as:

$$\mu = W\left[(1-u)^2 - (1-C_{\text{sol}})^2\right] \tag{9.2-24}$$

where W is the interaction parameter (J/mol) and C_{sol} (atom fraction) is the iron solubility limit.

With multiple (primary and forming) phases present, the chemical potential is defined as:

$$\mu = \sum_m \mu_m = \sum_m f_m W_m\left[(1-u)^2 - (1-C_{\text{sol}}^m)^2\right] \tag{9.2-25}$$

where f_m is the fraction of phase m present in the material.

When the solubility limit of a phase is exceeded, the supersaturation drives the growth of the new phase at the expense of the dissolution of the existing phase. The rate of phase growth under supersaturation is assumed to be temperature-dependent and modeled using the following equation:

$$\frac{\partial\eta_m}{\partial t} = k_{m0}\exp\left(-\frac{Q_m}{RT}\right) \tag{9.2-26}$$

where η_m is the phase fraction of phase m, k_{m0} is the phase growth kinetics constant (1/s), Q_m is the activation energy (J/mol), R is the universal gas constant, (J/mol/K), and T is the temperature (K).

9.2.2.6 Numerical Solution

Iron migration is solved in radial coordinates between the fuel centerline and clad outer surface. The chemical interaction starts only if there is chemical contact between the fuel and cladding. Diffusion is assumed to take place under the chemical potential difference between radial cells.

Using a finite volume approach and applying the divergence theorem, the following discretized equation is obtained for the iron diffusion through fuel and cladding:

$$\frac{d\overline{u}_i}{dt}\frac{r_{i+\frac{1}{2}}^2 - r_{i-\frac{1}{2}}^2}{2} = [ruM\nabla\mu]_{r_{i-\frac{1}{2}}}^{r_{i+\frac{1}{2}}} \tag{9.2-27}$$

where i is the radial cell, r is the radial location (m), u is the iron atom fraction, μ is the chemical potential of iron (J/mol), $M = \frac{D}{RT}$ is the *mobility*, D is the diffusion coefficient (m^2/s), R is the universal gas constant (J/mol/K), and T is temperature (K).

Growth of the forming phase at a radial node-i using explicit time discretization of Eq. (9.2-26) as follows:

$$\frac{\eta_{mi}^k - \eta_{mi}^{k-1}}{\Delta t} = k_{m0}\exp\left(-\frac{Q_m}{RT}\right) \tag{9.2-28}$$

where η_{mi}^k and η_{mi}^{k-1} are the phase fraction of mth phase for node-i at current and previous time steps, respectively, k_{m0} is the phase growth kinetics constant (1/s), Q_m is the activation energy (J/mol), R is the universal gas constant, (J/mol/K), and T is the temperature (K), Δt is the time step (s) .

Prior to the pre-transient irradiation, MFUEL assumes that there is no iron in the fuel and that the iron content in the cladding is uniform.

9.2.2.7 Boundary Conditions

$$\left.\frac{d\mu}{dr}\right|_{r=0} = 0 \tag{9.2-29}$$

$$\left.\frac{d\mu}{dr}\right|_{r=R} = 0 \tag{9.2-30}$$

where R is the clad outer surface.

9.2.2.8 Time Step Control

In order to assure numerical convergence, the time step for the iron migration model is selected as a function of the fuel centerline temperature as given in Eq. (9.2-31).

$$\Delta t_{iron} = min\left(\begin{cases} 5\ s & T_c \geq 950\ K \\ 25\ s & 900 \leq\ T_c < 950\ K \\ 50\ s & 850 \leq\ T_c < 900\ K \\ 100\ s & T_c < 850\ K \end{cases}, \Delta t_{MFUEL}\right) \tag{9.2-31}$$

where Δt_{MFUEL} is the MFUEL main time step, Δt_{iron} is the selected time step for the iron migration model, and T_c is the fuel centerline temperature (Kelvin). Section 9.3 describes MFUEL's main time step for steady state and transients.

9.2.2.9 Eutectic Formation

Above the fuel liquefaction temperature, eutectic formation and penetration into the cladding takes place via migration of actinides and iron at the fuel/cladding interface, resulting in liquefaction of the cladding. There are slow and rapid eutectic reactions. Ref. [9-7] experimentally shows that metallic fuel liquefaction at temperatures below 1080°C is associated with the $(U, Pu)_6Fe$ phase, and the solubility limit of plutonium in this phase makes the transition plutonium dependent. As a result, the liquefaction for U-10Zr is assumed to take place at 715°C. The liquefaction temperature drops to 650°C for the U-19Pu-10Zr fuel alloy [9-7]. Rapid eutectic formation is associated with melting of the $(U, Pu)Fe_2$ phase above 1080°C [9-8].

There are two different eutectic formation models incorporated as part of this development, an empirical and a mechanistic model. The mechanistic model is applicable only to the slow eutectic formation and builds on top of the iron migration model. The empirical model covers both slow and rapid eutectic formation.

9.2.2.9.1 Mechanistic Eutectic Model

In the mechanistic model, slow eutectic formation at the fuel/cladding interface is activated once a threshold temperature is reached. Beyond this temperature, the mechanistic model accounts for actinide diffusion and eutectic phase formation by solving Eq. (9.2-23) and Eq. (9.2-26), respectively, in addition to iron diffusion. As actinides and iron diffuse towards the eutectic interphase, the supersaturation drives the eutectic growth towards the cladding.

9.2.2.10 Numerical Solution

Actinide migration is solved by a finite volume approach and applying the divergence theorem. The following discretized equation is obtained for actinide transport from fuel to the fuel-cladding interface:

$$\frac{d\overline{u}_i}{dt}\frac{r_{i+\frac{1}{2}}^2 - r_{i-\frac{1}{2}}^2}{2} = [ruM\nabla\mu]_{r_{i-\frac{1}{2}}}^{r_{i+\frac{1}{2}}} \tag{9.2-32}$$

where i is the radial cell, r is the radial location (m), u is the actinide atom fraction, μ is the chemical potential of actinides, (J/mol), $M = \frac{D}{RT}$ is the *mobility*, and D is the diffusion coefficient, (m^2/s), R is the universal gas constant (J/mol/K), and T is temperature (K).

When the solubility limit of actinides is exceeded, the supersaturation drives the growth of the eutectic phase into the next cell at the expense of the dissolution of the cladding. The rate of phase growth under supersaturation is assumed constant using Eq. (9.2-28) but with the parameters corresponding to the eutectic model.

9.2.2.11 Boundary Conditions

$$\left.\frac{d\mu}{dr}\right|_{r=0} = 0 \tag{9.2-33}$$

$$\left.\frac{d\mu}{dr}\right|_{r=R} = 0 \tag{9.2-34}$$

where R is the clad outer surface (m).

9.2.2.11.1 Empirical Eutectic Model

The empirical model is based on Ref. [9-8] and given in Eq. (9.2-35). This is the only model applicable to rapid eutectic formation, however, it also covers the slow eutectic formation range.

$$E_{rate} = S_{Eu} \begin{cases} C_{eut}\left(1.1338 \times 10^{-9}\Delta T^3 - 2.1522 \times 10^{-7}\Delta T^2 \right. & \\ \left. +2.9265 \times 10^{-6}\Delta T + 9.22 \times 10^{-4}\right) & T_{low} < T < T_{high} \\ \left(\exp\left(22.847 - \frac{27624}{T}\right)\right) \times 10^{-6} & \text{else} \end{cases} \tag{9.2-35}$$

where T is Fuel Surface Temperature (K), $\Delta T = T - T_{peak}$ is the difference in temperature from the peak (K),E_{rate} is the eutectic formation rate (m/s), C_{eut} is a scaling factor which has been experimentally shown to depend on Pu content of the fuel, $T_{low} = 1353.15$ is the lowest rapid eutectic formation temperature (K), $T_{high} = 1506.15$ is the highest rapid eutectic formation temperature (K), $T_{peak} = 1388.15$ is the temperature at which peak eutectic formation rate occurs (K), and S_{Eu} is the user-defined sensitivity coefficient for the formation rate, **SCEUR**.

Similar to the lanthanide migration model, the eutectic layer is assumed not to bear any load in mechanical analysis.

9.2.2.12 Time Step Control

Since the eutectic model is active only during the transients, the time step for this model is equal to the MFUEL's main time step.

9.2.3 Fuel Swelling and Fission Gas Release

Diffusion of fission gas atoms takes place up to the point where they get trapped by a sink such as phase boundaries, grain boundaries, dislocations, existing bubbles, or open porosity. The growth of bubbles is driven by gas diffusion, bubble coalescence, and operating conditions. Post-irradiation examination of high smear density EBR-II metal fuels [9-9] indicates that a threshold gas swelling exists at approximately 10% gas swelling. Above the threshold gas swelling, fuel fracture and formation of interconnected open porosity network begins to take place. At this point, the gas in the fuel matrix or inside growing bubbles can be directly released to the plenum via interconnected porosity. After the formation of a significant amount of open porosity networks, gas swelling tends to saturate. However, solid fission product swelling continues. When hard contact (see Section 9.2.5) between fuel and cladding occurs, pressure sintering of the open porosity takes place to accommodate for solid fission product swelling.

Fission gas behavior models are based on the GRSIS algorithm developed by KAERI [9-10] and further improved in the FEAST project at MIT [9-11], [9-12]. The model takes a similar approach to the FEAST code [9-11], [9-12] but is extended to account for the transient behavior, including fuel temperatures above the solidus temperature. A constant atom number bubble-grouping scheme is adopted considering morphological variation within the fuel due to formation of different phase structures. Bubbles in $\alpha + \delta$ phase structure are treated as ellipsoidal whereas bubbles formed in $\beta + \gamma$ and single γ phases are spherical. Three groups of closed bubbles (1: small, 2: medium, 3: large) and three groups of open porosity bubbles (4-1: connected small, 4-2: connected medium, 4-3: connected large) are modeled. Based on experimental observations in TREAT tests and furnace tests, large bubbles, which are 125 times larger than the medium bubble group, only occur at temperatures above the solidus temperature.

Fission gas atoms are generated by fission, and then form (nucleate) new bubbles or diffuse into existing bubbles. The bubbles are assumed to nucleate uniformly in the metallic fuel matrix due to the fact that they nucleate at both the grain boundaries and the phase boundaries, which are randomly distributed inside the grain. The closed bubbles can grow by trapping the diffusing fission gas atoms within the fuel matrix.

Open bubbles (or open pores) are chains of closed bubbles which have connected to each other and extend to the external free surface. When a closed bubble-i becomes an open porosity, it is transformed into bubble-4i. When the fuel matrix swelling due to the closed bubbles reaches the threshold value, it is assumed that a certain fraction of the bubbles become interconnected and release their gas into the free volume (i.e., they become open porosity). Open porosity bubbles are assumed not to exist until the threshold gas swelling, 10%, has been reached. Once the threshold has been reached, it is assumed that 1% of the closed bubbles create the open porosity network and gas release begins to take place.

9.2.3.1 Governing Equations

The mechanistic model consists of a set of equations for gas atoms in the fuel matrix, closed bubbles, open porosity and released gas to the interconnected porosity, and constitutive models for several reactions such as gas bubble nucleation, gas diffusion coefficients, bubble coalescence, etc.

Total fuel swelling is due to gas bubbles, open porosity, and solid fission products. The solid fission product swelling rate is set to 1.5%/ at% burnup. The total swelling is given as follows:

$$S_0 = \frac{\Delta V}{V_0} = N_{b1}V_{\ 1} + N_{b2}V_2 + N_{b3}V_3 + N_{b41}V_1 + N_{b42}V_2 + N_{b43}V_3 + 0.015 \times bu \tag{9.2-36}$$

where Bu is the burnup (at %), S_0 is the total fuel swelling (-) with respect to initial volume, V_0 (m^3), N_{bi} is the density of closed bubble-i in the matrix (bubble/m^3), N_{b4i} is the density of porosity of type-i in matrix (bubble/m^3), and V_i is the volume of bubble-i (m^3/bubble-i).

The density of gas, closed bubbles, and open bubbles within the matrix is determined using:

Gas atom density in fuel matrix:

$$\frac{dC_g}{dt} = YF - (J_{g1} + J_{g2} + J_{g3} + J_{g4}) - J_{b_1,nucl} \tag{9.2-37}$$

Closed bubble density:

$$\begin{aligned}\rho_{g1}\frac{dN_{b1}}{dt} = (J_{b_1,nucl} + J_{g1} - (ab_{12} + ab_{13}) - (gab_{12} + gab_{21} + gab_{13} + gab_{31}) \\ -f_{12}\,(ab_{11} + gab_{11}) - ab_{14} - gab_{14})\end{aligned} \tag{9.2-38}$$

$$\begin{aligned}\rho_{g2}\frac{dN_{b2}}{dt} = (J_{g2} + ab_{12} + gab_{12} + gab_{21} + f_{12}\,(ab_{11} + gab_{11}) - (gab_{23} + gab_{32}) - \\ f_{23}\,(ab_{22} + gab_{22}) - ab_{23} - ab_{24} - gab_{24})\end{aligned} \tag{9.2-39}$$

$$\begin{aligned}\rho_{g3}\frac{dN_{b3}}{dt} = (J_{g3} + ab_{13} + ab_{23} + gab_{13} + gab_{31} + gab_{23} + gab_{32} + \\ f_{23}\,(ab_{22} + gab_{22}) - ab_{34} - gab_{34})\end{aligned} \tag{9.2-40}$$

Open bubble density:

$$\rho_{g1}\frac{dN_{b41}}{dt} = (ab_{14} + gab_{14}) + \rho_{g1}\epsilon_{sint}N_{b41} \tag{9.2-41}$$

$$\rho_{g2}\frac{dN_{b42}}{dt} = (ab_{24} + gab_{24}) + \rho_{g2}\epsilon_{sint}N_{b42} \tag{9.2-42}$$

$$\rho_{g3}\frac{dN_{b43}}{dt} = (ab_{34} + gab_{34}\ ) + \rho_{g3}\epsilon_{sint}N_{b43} \tag{9.2-43}$$

Total gas release density:

$$\frac{dC_{gr}}{dt} = J_{g4} + ab_{14} + ab_{24} + ab_{34} + gab_{14} + gab_{24} + gab_{34} \tag{9.2-44}$$

where C_g is the gas atom concentration in the fuel matrix (atoms/m^3), Y is the fission yield of gas atoms, assumed to be 0.25 atoms/fission, F is the fission density with respect to the as-fabricated volume (fission/s/m^3), J_{gi} is the gas diffusion rate to bubble-i (atoms/m^3/s), $J_{b_1,nucl}$ is the Bubble-1 nucleation rate (atoms/m^3/s), ab_{ij} is the transfer rate of bubble-i into bubble-j by bubble diffusion (atoms/m^3/s), gab_{ij} is the transfer rate of bubble-i into bubble-j by radial growth of bubble-i (atoms/m^3/s), ϵ_{sint} is the sintering strain rate on the fuel (1/s), $f_{i,i+1}$ is the transition probability of bubble-i into bubble-(i+1) due to collision, and ρ_{gi} is the constant atom number per bubble-i.

Equilibrium bubble volume is determined based on Van der Waal's equation:

$$\frac{V_{bei}}{\rho_{gi}} = B + \left[\left(\frac{\lambda\gamma}{kT}\right)\frac{1}{R_{ei}} + \frac{\sigma_h}{kT}\right]^{-1} \tag{9.2-45}$$

where:

$$\sigma_h = -\frac{\sigma_r + \sigma_\theta + \sigma_z}{3} \tag{9.2-46}$$

and B is the Van der Waals Parameter ($85 \times 10^{-30}\ m^3/atom$), $\gamma = 0.8$ is the surface tension (N/m) [9-13], $R_i = \left(\frac{3V_i}{4\pi}\right)^{1/3}$is the radius of bubble-i (m), σ_h is the hydrostatic stress, k= 1.3806×10^{-23} is the Boltzmann constant (J/K), and λ is the bubble correction factor. For spherical bubbles, the value of λ is 2 [9-14], but is reduced to 0.8 for ellipsoidal bubbles.

During pre-transient characterization, bubbles are assumed to be in equilibrium, $V_{bi} = V_{bei}$. During the transients, non-equilibrium is assumed between the bubbles and the fuel matrix. It is assumed that bubble volume changes towards the equilibrium volume as a function of fuel creep as follows:

$$\frac{dV_{bi}}{dt} = V_{bi}\epsilon_{cr}\left(V_{bei}\right) \tag{9.2-47}$$

where ϵ_{cr} is the fuel creep rate computed using pressure difference between matrix and bubble (1/s).

9.2.3.2 Numerical Solution

The fuel swelling and fission gas release model is a system of ordinary differential equations and is solved using a backwards Euler scheme. The system of ordinary differential equations is solved independently in every cell of the fuel. Prior to the pre-transient irradiation, MFUEL assumes there is no fission gas or porosity in the fuel. In order to ensure numerical stability, the time step used in this model is selected as a function of fuel temperature as given in Eq. (9.2-48). In single γ phase, larger diffusion coefficient dictates smaller time steps.

$$\Delta t_{fisg} = \min\left(\begin{cases} 200\ s & T < T_\gamma \\ 100\ s & T \geq T_\gamma \end{cases}, \Delta t_{MFUEL}\right) \tag{9.2-48}$$

where Δt_{MFUEL} is the MFUEL coarse time step (s), Δt_{fisg} is the selected time step for the fuel swelling and fission gas release model (s), T is the fuel temperature (K), and T_γ is the single gamma phase transition temperature (K). Section 9.3 describes MFUEL's main time step for steady state and transients.

9.2.4 Plenum Gas Pressure

The gas pressure in the free space of the fuel rod is computed assuming mechanical equilibrium and the ideal gas law. The initial fill gas, helium, is treated separately. The gas-filled plenum region and fuel's interconnected open porosity network is the free space accommodating the released gas from the fuel and helium gas.

The plenum gas pressure is computed as follows:

$$P_{plegas} = \frac{N_{FGRel} R T_{ave}}{N_{Av} V_{tot}} + P_0 \frac{T_{ave}}{T_R} \frac{V_R}{V_{tot}} \tag{9.2-49}$$

$$V_{tot} = V_R + \Delta V_{Plth} - \Delta V_{Fuexp} + \Delta V_{Fupor} + \Delta V_{CladInnerExp} - \Delta V_{NaExp} \tag{9.2-50}$$

$$T_{ave} = \frac{V_{tot} T_{plenum} + \sum_{j=1}^{mz} \sum_{i=1}^{nt} T_{fuel}(i,j) \times V_{porg}(i,j)}{V_{tot} + \sum_{j=1}^{mz} \sum_{i=1}^{nt} V_{porg}(i,j)} \tag{9.2-51}$$

where P_{plegas} is the plenum gas pressure (Pa), N_{FGRel} is the number of fission gas atoms released to the free space (-), N_{Av} is Avogadro's number (-/mol), R is the universal gas constant (J/mol/K), T_{ave} is the average temperature of the overall free space (K), V_{tot} is the total available free space (m^3), P_0 is the initial fill-gas pressure (Pa), T_R is the reference temperature corresponding to the initial fill-gas pressure (K), and V_R is the reference volume corresponding to the initial fill-gas volume (m^3).

Note that V_{tot} is determined based on the porosity of the fuel, expansion of the fuel and cladding, and relocation of the in-pin sodium (m^3), where ΔV_{Plth} is the plenum region thermal expansion volume compared to as-fabricated dimensions (m^3), ΔV_{Fuexp} is the fuel slug expansion (m^3), ΔV_{Fupor} is the gas-filled fuel open porosity volume (m^3), $\Delta V_{CladInnerExp}$ is the volume change due to clad inner radius expansion at fuel region (m^3), and ΔV_{NaExp} is the volume change due to in-pin sodium expansion (m^3). In Eq. (9.2-51), T_{plenum} is the plenum gas temperature(K), and $T_{fuel}(i,j)$ and $V_{porg}(i,j)$ are the fuel temperature (K) and gas filled porosity volume (m^3) at a given radial and axial node (i,j).

Plenum gas pressure is computed at every time step directly by solving Eq. (9.2-49), Eq. (9.2-50), and Eq. (9.2-51), giving the plenum pressure at each time step.

9.2.5 Fuel Anisotropic Axial Elongation

Metallic fuels show anisotropic fuel axial elongation behavior. This is due to fuel cracking and irradiation growth/grain boundary tearing [9-15]. After the fuel slug comes into soft contact with the cladding, further axial growth is restrained. Soft contact takes place when cracked fuel comes to radial contact with the cladding, whereas hard contact starts when the fuel swelling fills up all the cracks. Soft contact also activates the chemical contact between fuel and cladding. To incorporate this effect, the effective fuel slug radius is defined as follows:

$$r_{eff} = r_0 + dr_{slug} + dr_{crack} = r_{slug} + dr_{crack} \tag{9.2-52}$$

where r_0 is the as-fabricated fuel slug radius (m), dr_{slug} is the radial strain increment due to thermal expansion, elasticity, creep, and swelling (m), r_{slug} is the current radial dimension without cracks/tears (m), and dr_{crack} is the radius increment due to cracks and grain boundary tearing (m). Note that dr_{crack} is not a physical boundary. It is only used to find out the contact condition between the fuel and cladding.

Fuel/cladding contact condition can be divided into three intervals:

$r_{eff} < r_c$: No restraint by the cladding (no contact)

$r_{slug} < r_c < r_{eff}$: Axial restraint by the cladding (soft contact)

$r_{slug} = r_c$: Both radial and axial expansion restrained by the cladding (hard contact)

where r_c is the cladding radius.

dr_{crack} is modeled empirically as a function of the as-fabricated gap, plutonium content, and peak linear heat rate to fuel diameter ratio (W/m^2). The latter is a measure of temperature gradient:

$$dr_{crack} = f_{crack} \times \Delta r_{gap0} \tag{9.2-53}$$

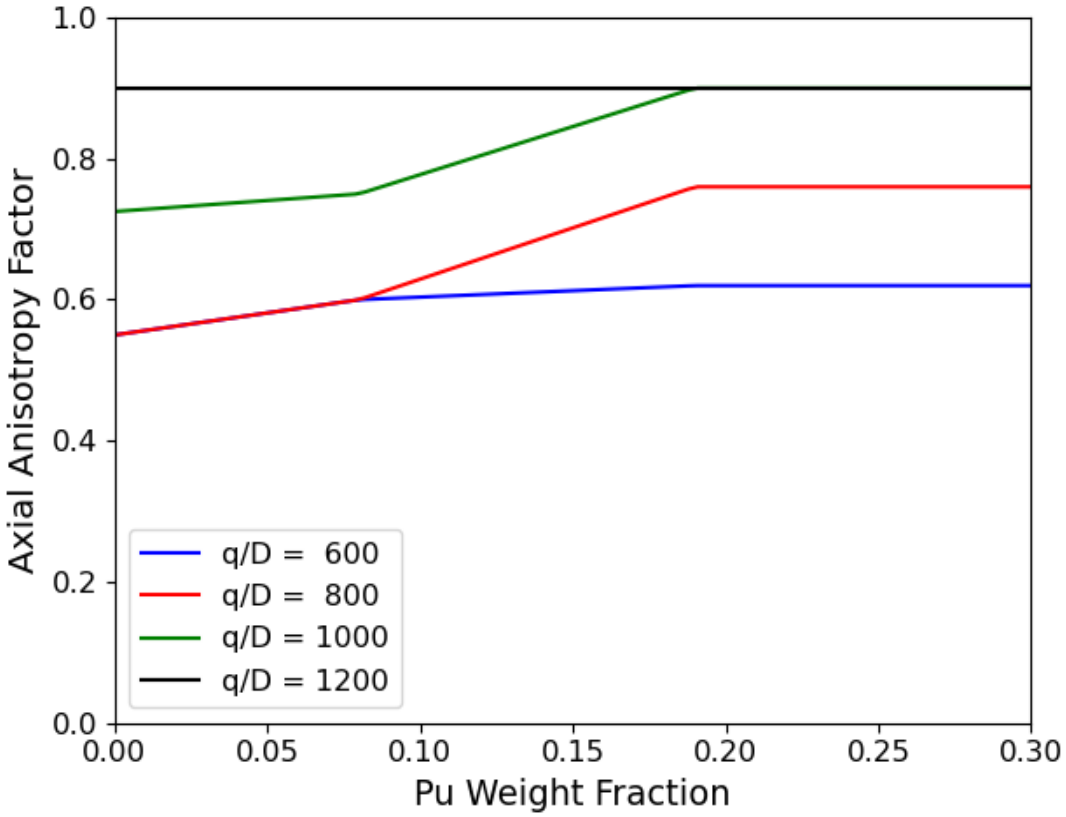


Figure 9.2.2: f_{crack} *empirical parameter as a function of Pu content and peak linear heat rate to fuel diameter ratio*

where Δr_{gap0} is the as-fabricated gap thickness (m) and f_{crack} is the axial anisotropy factor, modeled as a function of plutonium content and peak linear heat rate to fuel diameter ratio (W/m^2) as shown in Figure 9.2.2.

Fuel axial anisotropy factor is computed only once for each fuel channel during the steady state . Using the F parameter computed at peak power location within 0.5 at% peak burnup, the anisotropy factor is computed using the data presented in Table 9.2.2. The empirical model estimates at what fraction of the gap closure the soft contact will initiate. This value is assumed constant throughout the pre-transient characterization and transient simulations. The model is considered applicable for Pu contents between 0 to 26 wt%.

Table 9.2.2: Fuel axial anisotropy factor data used for bilinear interpolation

F/P	0.0	0.08	0.19
700	0.55	0.60	0.62
900	0.55	0.60	0.90
1100	0.90	0.90	0.90

Where Pu is plutonium weight fraction, and $F = \frac{Peak\ linear\ heat\ rate}{Initial\ Fuel\ slug\ diameter}$, W/cm^2

9.2.6 Fuel Pin Mechanical Analysis

The mechanical analysis model is a 1.5-dimensional axisymmetric reduced order model based on assumptions of linear elasticity and incremental strain theory. A plane strain approach is applied to compute the axial displacements and thin shell theory is assumed to compute the cladding stresses. Furthermore, the fuel creep rate is assumed be high enough to avoid any significant buildup of stress gradient within the fuel slug.

Up until contact with the cladding, the fuel can freely swell in the axial and radial directions. Due to cracking and irradiation growth/grain boundary tearing, the fuel first comes to a soft contact with the cladding (see Section 9.2.5). During soft contact, the interfacial stress between the fuel and cladding is low, yet it is assumed to be high enough to limit axial elongation of the fuel. During pre-transient characterization, it is assumed that soft contact prevents the fuel from expanding axially, but allows the fuel to expand radially towards the available free space. In transient analysis, fuel in soft contact is assumed to be able to expand radially and axially.

When hard contact exists between the fuel and cladding, interfacial stresses build up to limit the fuel movement both axially and radially. As a result, the fuel slug becomes fully constrained by the cladding.

When the default model is used (**IEUTOPT** = 0 or 1), in the case of eutectic formation between the fuel and cladding, in the absence of hard contact at any axial node above, the soft and liquified fuel is allowed to expand primarily in the axial direction in a frictionless manner. When **IEUTOPT** = 2 or 3 options are selected, pore sintering upon eutectic formation is allowed. These options lead to less aggressive axial expansion.

Each time step, the model computes the axial and radial displacements of the fuel and cladding based on the contact status and the strain components. Radial displacement is computed as:

$$u_r(r) = \epsilon_\theta R_{in} + \int_{R_{in}}^{r} \epsilon_\theta dr \tag{9.2-54}$$

where u_{cr} is the radial displacement as a function of radial position (m), R_{in} is the inner radius of the structure (m), and ϵ_θ is the circumferential strain (m/Δm).

Axial displacement is computed as:

$$\epsilon_z = \frac{\partial u_z}{\partial z} \tag{9.2-55}$$

$$\Delta u_z(r) = \epsilon_z(r)\Delta z \tag{9.2-56}$$

$$\Delta\overline{u}_z = \frac{1}{A}\int_A \Delta u_z dA \tag{9.2-57}$$

$$\overline{u}_z(i) = \sum_{j=1}^{i} \Delta\overline{u}_z(j) \tag{9.2-58}$$

where Δu_z is the radial displacement as a function of radial position at a given axial node (m), ϵ_θ is the axial strain (m/Δm), Δz is axial height at a given node (m), A is the cross-sectional area (m^2), $\Delta\overline{u}_z$ is the average axial displacement at a given axial node (m), and $\overline{u}_z(i)$ is the cumulative axial displacement at a given axial node I (m).

Strain is incremented based on the state of swelling, creep, thermal expansion, plenum pressure, and coolant pressure. Strain components are grouped into elastic and nonelastic contributions. Elastic and nonelastic strain components for fuel and cladding are conditional on the state of contact between the fuel and cladding. The strain on the fuel and cladding is described as:

$$\epsilon_r = \epsilon_{er} + \epsilon_{nr} \tag{9.2-59}$$

$$\epsilon_\theta = \epsilon_{e\theta} + \epsilon_{n\theta} \tag{9.2-60}$$

$$\epsilon_z = \epsilon_{ez} + \epsilon_{nz} \tag{9.2-61}$$

where:

$$\epsilon_{er} = \frac{1}{E}\left[\sigma_r - \nu\left(\sigma_\theta + \sigma_Z\right)\right] \tag{9.2-62}$$

$$\epsilon_{e\theta} = \frac{1}{E}\left[\sigma_\theta - \nu\left(\sigma_r + \sigma_Z\right)\right] \tag{9.2-63}$$

$$\epsilon_{ez} = \frac{1}{E}\left[\sigma_z - \nu\left(\sigma_r + \sigma_\theta\right)\right] \tag{9.2-64}$$

$$\epsilon_{nr} = \alpha(T - T_0) + \epsilon_s + \epsilon_r^c \tag{9.2-65}$$

$$\epsilon_{n\theta} = \alpha(T - T_0) + \epsilon_s + \epsilon_\theta^c \tag{9.2-66}$$

$$\epsilon_{nz} = \alpha(T - T_0) + \epsilon_s + \epsilon_z^c \tag{9.2-67}$$

where $\sigma_r, \sigma_\theta, \sigma_z$ are the radial, azimuthal, and axial stress (Pa), $\epsilon_r, \epsilon_\theta, \epsilon_z$ are the radial, azimuthal, and axial total strain components (m/Δm), $\epsilon_{er}, \epsilon_{e\theta}, \epsilon_{ez}$ are the radial, azimuthal, and axial elastic strain components (m/Δm), $\epsilon_{nr}, \epsilon_{n\theta}, \epsilon_{nz}$ are the radial, azimuthal, and axial non-elastic strain components (m/Δm), E is the Elastic Modulus (Pa), ν is the Poisson's ratio (-), α is the linear thermal expansion coefficient (m/Δm-K), T is the material temperature (K), T_0 is the reference temperature (K), ϵ_s is the swelling strain (m/Δm), and ϵ_r^c, ϵ_θ^c, ϵ_z^c are the radial, azimuthal, and axial creep strain (m/Δm).

During pre-transient characterization, soft contact conditions are assumed to restrict movement in the axial direction. To account for this, the axial non-elastic strain components of the fuel are transferred to the radial and azimuthal directions as follows:

$$\epsilon_r = \epsilon_{er} + \epsilon_{nr} + 0.5\epsilon_{nz} \tag{9.2-68}$$

$$\epsilon_\theta = \epsilon_{e\theta} + \epsilon_{n\theta} + 0.5\epsilon_{nz} \tag{9.2-69}$$

$$\epsilon_z = \epsilon_{ez} \tag{9.2-70}$$

Similarly, when the fuel and cladding are subject to hard contact conditions and all of the fuel slugs above the current elevation have formed a eutectic, the default model (IEUTOPT = 0 or 1) assumes that strain in the axial

direction is dominant. To account for this, the radial and azimuthal non-elastic strain components of the fuel are transferred to the axial directions as follows:

$$\epsilon_r = \epsilon_{er} \tag{9.2-71}$$

$$\epsilon_\theta = \epsilon_{e\theta} \tag{9.2-72}$$

$$\epsilon_z = \epsilon_{ez} + \epsilon_{nz} + \epsilon_{nr} + \epsilon_{n\theta} \tag{9.2-73}$$

Users may skip this model and use Eq. (9.2-41) through Eq. (9.2-43) if **IEUTOPT** is set to 2 or 3.

9.2.6.1 Fuel Stress

Fuel stress is computed using a reduced order model. The reduced order model assumes no stress gradient within the fuel or the cladding. Prior to hard contact, the fuel stress distribution at a given axial location is given as follows:

$$\sigma_{fr} = \sigma_{f\theta} = -P_{Plgas} \tag{9.2-74}$$

$$\sigma_{fz} = -(P_{Plgas} + P_{fw}) \tag{9.2-75}$$

where $\sigma_{fr}, \sigma_{f\theta}, \sigma_{fz}$ are the fuel radial, azimuthal, and axial stress (Pa), P_{Plgas} is the plenum gas (Pa), and P_{fw} axial force due to fuel weight on the fuel surface (Pa).

Upon formation of the hard contact between the fuel and cladding, the primary mechanism balancing any fuel axial expansion is porosity sintering or hot pressing, which is assumed to be isotropic. The interfacial stress is computed such that the fuel outer and clad inner displacements are aligned as shown in Eq. (9.2-76).

$$u_{fr}|_{outer} \cong u_{cr}|_{inner} \& u_{fz}|_{outer} \cong u_{cz}|_{inner} \tag{9.2-76}$$

$$\sigma_{fr} = \sigma_{f\theta} = \sigma_{fz} = \sigma_{contact} \tag{9.2-77}$$

Contact stress is computed in a stepwise manner such that every time step its value is allowed to increase or decrease incrementally towards the equilibrium value. Default stress step size is 0.1 MPa. However, if the expansion or contraction rate requires larger step size, the stress step is increased to 1 MPa. This relaxation ensures numerical convergence without disrupting the accuracy.

9.2.6.2 Clad Stress

Cladding stress is assumed using a thin shell theory based on the applied stress at the inner and outer surface. The thin shell theory equations used to calculate the cladding principal stress are:

$$\sigma_{cr} = \frac{\sigma_{inner\ R} + \sigma_{outer\ R}}{2} \tag{9.2-78}$$

$$\sigma_{c\theta} = -\frac{(\sigma_{inner\ R} - \sigma_{outer\ R})R_c}{\Delta r} \tag{9.2-79}$$

$$\sigma_{cz} = -\frac{(\sigma_{inner\ R} - \sigma_{outer\ R})R_c}{2 \times \Delta r} \tag{9.2-80}$$

where $\sigma_{cr}, \sigma_{c\theta}, \sigma_{cz}$ is the clad radial, hoop, and axial stress (Pa), $\sigma_{inner\ R}$ is the contact stress between fuel and cladding (Pa), $\sigma_{outer\ R}$ is the coolant pressure, and Δr is the current active clad thickness (m), where the clad wastage has been subtracted as it is assumed to not bear any load.

9.2.6.3 Fuel Swelling Strain

Fuel swelling strain is the largest in magnitude compared to other strain components (see Section 9.2.3). In order to be compliant with the incremental strain theory and the reduced order modeling approach, fuel swelling strain is first radially averaged for a given time step, and the average radial value is used to determine the axial displacements, shown in Eq. (9.2-81):

$$\epsilon_s = \frac{\overline{S}}{3} \tag{9.2-81}$$

$$\overline{S} = \frac{1}{V}\int_V S\, dV = \frac{1}{V}\int_V \frac{S_0}{(1+S_0+3\alpha(T-T_R))} dV \tag{9.2-82}$$

where ϵ_s is the average linear swelling strain (-), $\overline{S}$ is the radial average volumetric swelling with respect to the current fuel dimensions (-), S_0 is the local fuel swelling with respect to the fresh fuel dimensions (-), S is the local fuel swelling with respect to the current fuel dimensions (-), and V is the fuel volume of the axial zone (m^3).

Following the radial displacement of the nodes using Eq. (9.2-54), the radial positions are corrected using the local swelling strain:

$$A_i^c = A_i \frac{1+S}{1+\overline{S}} \tag{9.2-83}$$

$$R_i = \sqrt{A_{i,c}\frac{\sum_{j=1}^{i} A_j}{\sum_{j=1}^{i} A_j^c} + R_{i-1}^2} \tag{9.2-84}$$

where A_i is the node area (m^2), A_i^c is the corrected node area accounting for difference between average and local swelling strain (m^2), and R_i *and* R_{i-1} are the radial location for node-i and node-(i-1), respectively (m).

9.2.6.4 Clad Swelling Strain

For a given axial node, clad swelling strain is computed using the average clad temperature, neutron flux, and neutron fluence. In order to comply with the incremental strain theory approach, the incremental swelling strain at a given time step is adjusted to reflect the current volume:

$$\epsilon_s = \frac{\overline{S_0}}{3\left(1+\overline{S_0}+3\alpha(T-T_R)\right)} \tag{9.2-85}$$

$$\overline{S_0} = \int_{-\infty}^{t} f(T,\phi,\psi)dt$$

where T is the temperature at the clad midwall (K), ϕ is the neutron flux (#/cm^2-s), ψ is the neutron fluence (#/cm^2), ϵ_s is the linear incremental swelling strain with respect to the current volume (-), and $\overline{S_0}$ is the volumetric swelling strain with respect to the as-fabricated volume (-).

The volumetric swelling rate for SS316, D9, HT9, and 15-15Ti are defined as

SS316

$f_{SS316}(T,\phi,\psi)$ is defined in Eq. (8.3-65).

D9

$$f_{D9}(T,\phi,\psi) = \begin{cases} 0.0 & \text{IRATE} = -2 \\ f_{D9,0}(T,\phi,\psi) & \text{IRATE} = 0 \\ f_{D9,1}(T,\phi,\psi) & \text{IRATE} = 1 \end{cases} \tag{9.2-86}$$

$$f_{D9,0}(T,\phi,\psi) = \begin{cases} 0.002\phi D_c & T < 723 \text{ K and } D_c \cdot \psi \geq 55 \\ 0.005\phi D_c & T \geq 723 \text{ K and } D_c \cdot \psi \geq 60 \\ 0 & \text{else} \end{cases} \tag{9.2-87}$$

where D_c is the user defined dose to neutron fluence ratio, **CDOSECONV** [9-31].

$$f_{D9,1}(T,\phi,\psi) = 0.01\phi \left(\frac{R(T)}{1+\exp\left(\alpha\left(\tau-\psi\right)\right)} + 0.534\exp\left(-3\psi\right) \right) \tag{9.2-88}$$

$$R(T) = \exp\left(0.471 - 0.003B + 0.179B^2 - 0.077B^3 - 1.78B^4\right)$$

$$B = 0.01\left(T - 773.15\right)$$

where $\alpha = 0.6 \times 10^{22}$ is the curvature parameter (#/cm^2), and $\tau = 16.1 \times 10^{22}$ is the incubation parameter (#/cm^2) [9-22].

HT9

$$f_{HT9}(T,\phi,\psi) = \begin{cases} 0.0 & \text{IRATE} = -2 \\ f_{HT9,0}(T,\phi,\psi) & \text{IRATE} = 0 \\ f_{HT9,1}(T,\phi,\psi) & \text{IRATE} = 1 \end{cases} \tag{9.2-89}$$

$$f_{HT9,0}(T,\phi,\psi) = \begin{cases} 0.0 & D_c \cdot \psi < 100 \\ A(T)D_c\phi & \text{else} \end{cases} \tag{9.2-90}$$

where $A(T)$ is linearly interpolated from the data presented in Table 9.2.3 and D_c is the user defined dose to neutron fluence ratio, **CDOSECONV** [9-11] [9-34].

Table 9.2.3: Selected reference point for HT9 void swelling rate [9-34]

Temperature (K)	Value
0.0	-1.24782×10^{-4}
673.0	1.0×10^{-4}
723.0	8.33×10^{-5}
773.0	5.0×10^{-5}
823.0	2.5×10^{-5}
873.0	1.25×10^{-5}
873.0	0.0

$$f_{HT9,1}(T,\phi,\psi) = 0.01\phi \left(\frac{R(T)}{1+\exp\left(\alpha\left(\tau-\psi\right)\right)} + 0.015\exp\left(-0.1\psi\right) \right) \tag{9.2-91}$$

$$R(T) = 0.085\exp\left(-1 \times 10^{-4}\left(T - 673.15\right)^2\right)$$

where $\alpha = 0.75 \times 10^{22}$ is the curvature parameter (#/cm^2), and $\tau = 14.2 \times 10^{22}$ is the incubation parameter (#/cm^2) [9-22].

15-15Ti

$f_{1515Ti}(T,\phi,\psi)$ is defined in Section 8.7.17.3.

User Defined

In addition to using the built-in volumetric swelling rates, users may define a custom volumetric swelling rate using **iCSwellID**.

$$f_{custom}(T, \phi, \psi) = \phi D_c R(T, D_c \cdot \psi) \tag{9.2-92}$$

where D_c is the user defined dose to neutron fluence ratio, **CDOSECONV**, and $R(T, D_c \cdot \psi)$ is the user-defined function, **iCSwellID**, with temperature (K) and cladding dose (dpa) as the independent variables.

9.2.6.5 Fuel Creep Strain

Metal fuel typically operates with a high creep rate due to its porous nature and low fuel solidus temperature. As a result, reduced order mechanical analysis models are more appropriate than full elastoplastic solutions with Finite Element (FE) or Finite Volume (FV) approaches considering computational efficiency versus accuracy. MFUEL accounts for the creep strain as the driving force for the pore sintering, which is an essential component to drive the fuel and cladding displacements, as well as stresses. Therefore, creep terms within the fuel, with the exception of hot pressing, are set to zero.

$$\epsilon_r^c = \epsilon_\theta^c = \epsilon_z^c = 0.0 \tag{9.2-93}$$

Hot pressing strain is defined as a function of equivalent creep strain [9-9]. If the average stress on fuel matrix is greater than the gas pressure, the sintering strain is non-zero. Otherwise, sintering rate is set to zero. Moreover, in order to account for fission-driven creep, the minimum fuel temperature used to compute the creep rate is set to 800 K. This is consistent with transition from thermal to radiation enhanced diffusion at about half of the fuel solidus temperature [9-14]:

$$\epsilon_{sint} = \alpha_p \epsilon_{eq} \tag{9.2-94}$$

$$\epsilon_{eq} = \begin{cases} \left(5 \times 10^3 \sigma_{hp} + 6\sigma_{hp}^{4.5}\right) e^{-\frac{26170}{T}}, & phase \neq Gamma \\ 0.08\sigma_{hp}^3 e^{-\frac{14350}{T}}, & phase = Gamma \end{cases}$$

$$\sigma_{hp} = \begin{cases} 0.0, & -\frac{\sigma_r+\sigma_\theta+\sigma_z}{3\times10^6} \leq P_{plegas} \\ \left|-\frac{\sigma_r+\sigma_\theta+\sigma_z}{3\times10^6} - \frac{P_{plegas}}{1\times10^6}\right|, & -\frac{\sigma_r+\sigma_\theta+\sigma_z}{3\times10^6} > P_{plegas} \end{cases}$$

$$\alpha_p = \begin{cases} 0.0, & V_4 \leq 0.0 \\ \frac{C}{6}\left(\frac{V_4}{0.1}\right)^{1.5}, & 0.0 < V_4 < 0.1 \\ \frac{C}{6}, & V_4 > 0.1 \end{cases}$$

$$V_4 = N_{b41} V_1 + N_{b42} V_2 + N_{b43} V_3$$

where T is the temperature of the fuel (K), ϵ_{eq} is the equivalent strain rate (-), $\sigma_r, \sigma_\theta, \sigma_z$ are the principle stresses in r, θ, z directions, respectively (Pa), V_4 is the open porosity swelling (m^3) (see Section 9.2.3), α_p is the pore compressibility factor (-), and C is the empirical constant, set to 10 (-).

If the hydrostatic stress is higher than the fuel porosity sintering yield strength, pore sintering is computed using yield conditions (Section 9.8.5).

9.2.6.6 Clad Creep Strain

Clad creep strain primarily consists of irradiation and thermal components. The creep strain in r, θ, and z direction is computed using an equivalent strain and stress:

$$\epsilon_r^c = \frac{\epsilon_{eq}}{\sigma_{eq}} \left(\sigma_r - 0.5\left(\sigma_\theta + \sigma_z\right)\right) \Delta t \tag{9.2-95}$$

$$\epsilon_\theta^c = \frac{\epsilon_{eq}}{\sigma_{eq}} (\sigma_\theta - 0.5\left(\sigma_r + \sigma_z\right)) \Delta t$$

$$\epsilon_z^c = \frac{\epsilon_{eq}}{\sigma_{eq}} (\sigma_z - 0.5\left(\sigma_r + \sigma_\theta\right)) \Delta t$$

$$\epsilon_{eq} = g\left(T, \phi, \sigma_{eq}\right) + S_{tc} h\left(T, t, \sigma_{eq}\right)$$

$$\sigma_{eq} = \frac{1}{2}\left(\left(\sigma_r - \sigma_\theta\right)^2 + \left(\sigma_r - \sigma_z\right)^2 + \left(\sigma_\theta - \sigma_z\right)^2\right)^{0.5}$$

where T is the temperature at the inner surface of the cladding (K), ϕ is the neutron flux (#//cm^2-s), ϵ_{eq} is the equivalent creep strain rate (1/s), σ_{eq} is the equivalent stress rate (Pa) , $g\left(T, \phi, \sigma_{eq}\right)$ is an empirical function capturing the irradiation creep strain rate, and $h\left(T, t, \sigma_{eq}\right)$ is an empirical function capturing the thermal creep strain rate, and S_{tc} is a user-defined sensitivity coefficient for thermal creep strain rate, **SCCTC**.

The empirical functions capturing irradiation creep strain rate for SS316, D9, HT9, and 15-15Ti are defined as

SS316 and **D9**

$$g_{SS316}\left(T, \phi, \sigma_{eq}\right) = g_{D9}\left(T, \phi, \sigma_{eq}\right) = 0.01 \times A(T) \times \phi \times D_c \times \left(\sigma_{eq} \times\ 10^{-6}\right) \tag{9.2-96}$$

$$A(T) = \begin{cases} 2 \times 10^{-6} & T \leq 723.15\ K \\ 2 \times 10^{-6} + \frac{\left(3 \times 10^{-7} - 2 \times 10^{-6}\right)}{773.15 - 723.15}\left(T - 723.15\right) & 723.15 < T \leq 773.15 \\ 3 \times 10^{-7} & T >\ 773.15 \end{cases}$$

where D_c is the user defined dose to neutron fluence ratio, **CDOSECONV** [9-31].

HT9

$$g_{HT9}\left(T, \phi, \sigma_{eq}\right) = 0.01\left(1.83 \times 10^{-4} + 2.59 \times 10^{14} \exp\left(-\frac{7.3 \times 10^4}{RT}\right) \times \phi \times \left(\sigma_{eq} \times 10^{-6}\right)^{1.3}\right) \tag{9.2-97}$$

where R is the universal gas constant (cal/mol/K) [9-22].

15-15Ti

$$g_{1515Ti}\left(T, \phi, \sigma_{eq}\right) = 0.75 \times A(T) \times \phi \times D_c \times \left(\sigma_{eq} \times\ 10^{-6}\right) \tag{9.2-98}$$

where $A(T)$ is linearly interpolated from the data presented in Table 9.2.4 and D_c is the user defined dose to neutron fluence ratio, **CDOSECONV** [8-40].

Table 9.2.4: Selected reference point for 15-15Ti irradiation creep [8-40]

Temperature (K)	Value
673.15	0.6×10^{-6}
713.15	3.2×10^{-6}
743.15	2.3×10^{-6}
773.15	0.5×10^{-6}
800.15	1.0×10^{-6}
898.15	1.0×10^{-6}

User Defined

In addition to using the built-in irradiation creep rates, users may define a custom volumetric swelling rate using **iCIrrCreepID**.

$$f_{custom}(T,\phi,\psi) = A(T) \times \phi \times D_c \left(\sigma_{eq} \times 10^{-6}\right) \tag{9.2-99}$$

where D_c is the user defined dose to neutron fluence ratio, **CDOSECONV**, and $A(T)$ is the user-defined function, **iCIrrCreepID**, with temperature (K) as the independent variables.

The empirical functions capturing thermal creep strain rate for SS316, D9, HT9, and 15-15Ti are defined as

SS316 and **D9**

$$h_{SS316}\left(T,t,\sigma_{eq}\right) = h_{D9}\left(T,t,\sigma_{eq}\right) = \varepsilon_{OS} \times \left(\frac{\sigma_{eq}}{(20000 - 9.12 \times T) \times (92000 - 40.2 \times T)}\right)^m \times \exp\left(-\frac{Q_r}{T}\right) \tag{9.2-100}$$

where ε_{OS} is 38633 (1/s), m is 5.35, Q_r is 1.062e+14 K.

HT9

$$h_{HT9}\left(T,t,\sigma_{eq}\right) = \left(\varepsilon_{pHT9} + \varepsilon_{sHT9} + \varepsilon_{tHT9}\right) \times 0.01 \tag{9.2-101}$$

$$\varepsilon_{pHT9} = \left[C_1 \times EXP(-\frac{Q_1}{RT}) \times (\sigma_{eq} \times 10^{-6}) + C_2 \times EXP(-\frac{Q_2}{RT}) \times (\sigma_{eq} \times 10^{-6})^4 + C_3 \times EXP(-\frac{Q_1}{RT}) \times (\sigma_{eq} \times 10^{-6})^{0.5}\right] \times EXP(-C_4 \times t) \times C_4$$

$$\varepsilon_{sHT9} = C_5 \times EXP(-\frac{q4}{RT}) \times (\sigma_{eq} \times 10^{-6})^2 + C_6\ EXP(-\frac{q5}{RT}) \times (\sigma_{eq} \times 10^{-6})^5$$

$$\varepsilon_{tHT9} = \begin{cases} C_{HT9Creep} \times 4 \times C_7 \times EXP\left(-\frac{Q_6}{RT}\right) \times \left(\sigma_{eq} \times 10^{-6}\right)^{10} \times t^3 & T \leq 1200\ K \\ 0 & T > 1200\ K \end{cases}$$

where R is gas constant (cal/mol/K), t is time (s) and all remain coefficients are defined in Table 9.2.5 [9-30].

Table 9.2.5: HT9 thermal creep parameters

C_1	13.4
C_2	8.43e-03
C_3	4.08e+18
C_4	1.6e-06
C_5	1.17e+9
C_6	8.33e+9
C_7	9.53e+21
Q_1	15027
Q_2	26451
Q_3	89167
Q_4	83142
Q_5	108276
Q_6	282700
$C_{HT9Creep}$	0.15

15-15Ti

$h_{1515Ti}\left(T,t,\sigma_{eq}\right)$ is defined in Section 8.7.17.1.

9.2.6.7 Numerical Solution

At each time step, the model updates the contact state between the fuel and cladding and contact pressure using previous time step nodal positions. Then the model determines the elastic and non-elastic strain components in each radial node of the fuel and cladding.

Incremental strain components, $\Delta\epsilon^t_{r,i}$, $\Delta\epsilon^t_{\theta,i}$, $\Delta\epsilon^t_{z,i}$, for radial node i are computed as follows:

$$\Delta\epsilon^t_{r,i} = \epsilon^t_{r,i} - \epsilon^{t-1}_{r,i} \tag{9.2-102}$$

$$\Delta\epsilon^t_{\theta,i} = \epsilon^t_{\theta,i} - \epsilon^{t-1}_{\theta,i} \tag{9.2-103}$$

$$\Delta\epsilon^t_{z,i} = \epsilon^t_{z,i} - \epsilon^{t-1}_{z,i} \tag{9.2-104}$$

9.2.6.8 Fuel displacements

The radial and axial fuel displacements are computed using linear elasticity and strain/displacement relationships:

$$u^t_{r,i} = \sum_{i'=1}^{i} \Delta\epsilon^t_{\theta,i'}(R^t_{i'+1} - R^t_{i'}) \tag{9.2-105}$$

$$\Delta u^t_{z,j} = \frac{1}{A_f}\sum_{i=1}^{NT} \Delta\epsilon^t_{z,i}\Delta z_j \Delta A_{fi} \tag{9.2-106}$$

$$\overline{u}^t_{z,j} = \sum_{j'=1}^{j} \Delta u^t_{z,j'} \tag{9.2-107}$$

where $u^t_{r,i}$ is the radial displacement at node i at time t, $\Delta u^t_{z,i}$ is the axial displacement computed using plain strain assumption, ΔA_{fi} is the cross-sectional area of node i, A_f is the total fuel cross-sectional area, and NT is the number of radial nodes in the fuel. The cumulative value of the axial displacements is computed by Eq. (9.2-58) for axial node j.

If the eutectic forms between the fuel and cladding at hard contact condition, contact pressure is set to the plenum pressure. As a result the constraint keeping the fuel and cladding interface is lost. To correct for this, if the fuel boundary exceeds the clad boundary, the following correction is applied to the computed radial node positions:

$$area_ex = \frac{R^2_{ci}}{R^2_{fo}} \tag{9.2-108}$$

$$R_i = \sqrt{R^2_{i0} \times area_ex} \tag{9.2-109}$$

Where $area_ex$ is the area fraction that fuel will be contracted radially if the conditions are met (m^2), R_{ci} and R_{fo} are clad inner and fuel outer radius (m). R_{i0} and R_i are the fuel radial node positions (m) for previous and current time steps, respectively.

9.2.6.9 Clad Displacements

The radial and axial clad displacements are computed using linear elasticity and strain/displacement relations:

$$u^t_{cr,i} = \Delta\epsilon^t_{\theta,NE} R^t_{NE} + \sum_{i'=NE}^{NE+i-1} \Delta\epsilon^t_{\theta,i'} (R^t_{i'+1} - R^t_{i'}) \tag{9.2-110}$$

$$\Delta u^t_{cz,j} = \frac{1}{A_c} \sum_{i'=NE}^{NE+n-1} \Delta\epsilon^t_{z,i'} \Delta z_j \Delta A_{ci'} \tag{9.2-111}$$

$$\overline{u}^t_{z,j} = \sum_{j'=1}^{j} \Delta u^t_{z,j'} \tag{9.2-112}$$

where $u^t_{cr,i}$ is the radial displacement of node i (m) at time t, $\Delta u^t_{cz,j}$ is the axial displacement (m) computed using plain strain assumption, ΔA_{ci} is the cross-sectional area of node i(m^2), A_c is the total clad cross-sectional area(m^2), R^t_{NE} is the clad inner radius (m) at time t, and n is the number of nodes in the cladding. The cumulative value of the axial displacement is computed by Eq. (9.2-58) for axial node j.

9.2.7 In-Pin Sodium Treatment

The in-pin sodium bond between the fuel and the cladding will relocate during irradiation. As the fuel swells, the bond sodium is primarily pushed to the top of the fuel column, reducing the available gas volume. In addition, as the interconnected porosity forms, bond sodium can infiltrate into some of the open porosity, leading to significant changes in fuel thermal conductivity. The sodium infiltration primarily takes place at the fuel outer region where $\alpha + \zeta + \delta$ phases exist and larger lenticular bubbles and cracks form. In $\beta + \zeta + \delta$ phase, bubbles are rather small, hindering sodium infiltration.

The porosity fraction that is filled by sodium is computed using the following relations in the $\alpha + \zeta + \delta$ phase region for fuel nodes that are radially located beyond 60% of the fuel radius:

$$P_{Na,fr} = \begin{cases} 0.6 & contact \neq hard \\ \max(0.3,\ 0.6 - 5(B - B_0)) & contact = hard \end{cases} \tag{9.2-113}$$

where:

$P_{Na,fr}$: Fraction of the fuel porosity that is filled by sodium. Minimum allowed value is 0.3 for $\alpha + \zeta + \delta$ phase and for the fuel radial location that is greater than 60% of the fuel radius

B: Burnup (atom fraction)

B_0: Burnup at which the fuel and clad hard contact initiated (atom fraction)

If the fuel is not $\alpha + \zeta + \delta$ phase or the radial location is less than 60% of the fuel radius, it is assumed there is no sodium infiltration. This assumption is based on experimental data given in Ref. [9-16].

At each time step during pre-transient characterization, the movement of sodium within the pin is updated as a function of phases present in the fuel, burnup, and the width of the fuel/cladding gap in an explicit manner. This information is used to compute the fuel thermal conductivity and plenum pressure. In transient, the sodium infiltrated into fuel porosity is not updated and assumed immobile.

9.2.8 Clad Outer Corrosion

Although stainless steel and sodium coolant have excellent compatibility, the minor corrosion that takes place is modeled empirically based on Ref. [9-17] using the following equation:

$$\frac{dr}{dt} = 3.310^{-6} \exp\left(-\frac{133031.4}{RT}\right) \tag{9.2-114}$$

where $\frac{dr}{dt}$ is the corrosion rate (m/s), R is the universal gas constant (J/mol/K), T is the clad outer temperature (K).

The corrosion thickness is solved at each axial node using a backward Euler method.

9.2.9 CDF-Based Clad Creep Rupture

Creep-induced nucleation and diffusion growth of the grain boundary cavities are the primary failure mechanisms of commercial steels being used as the cladding material. The mechanism is called "intergranular creep fracture." The complex phenomenon shows strong dependency on the present microstructure. The carbide/sulfide particles at the grain boundaries and their precipitation/coarsening or dissolution behavior, grain size, dislocation loop density, dislocation dynamics, crystallographic orientation, and diffusivity are the essential parameters affecting the progression of event. Furthermore, history of the operation during normal operation, anticipated transients, or accident scenarios could also affect the cavity dynamics by large. If excessively thick and brittle phases form at the clad inner surface, the stress concentrations may arise and present additional components limiting the life of the fuel pin and complicating the problem further.

The high complexity of the underlying physics has forced engineers to adopt simple empirical models to predict the clad failure, such as the Cumulative Damage Fraction (CDF) model. The CDF model uses the available clad failure data generated under constant stress and constant temperature conditions. Rupture time is predicted by the following equation when CDF reaches unity:

$$CDF = \int_0^{t_f} \frac{dt}{t_r\left(\sigma_\theta, T\right)} < 1 \tag{9.2-115}$$

where t_r is the rupture time (hr), t_f is the final time (hr), σ_θ is the hoop stress (Pa), and T is the temperature (K). Note that time to rupture correlations are derived from constant stress experiments whereas in simulation the magnitude of stress may change.

The SS316, D9, and HT9 CDF models are described below. The CDF model for 15-15Ti is defined in Section 8.7.17.4.

9.2.9.1 HT9 CDF Model

The HT9 CDF model determines the rupture time based on the temperature and stress. At low temperatures, T <= 973.15 K, the rupture time is determined using:

$$\log t_r = \begin{cases} \log t_{rs} & \sigma_\theta \le \sigma_{ss} \\ \frac{\sigma_\theta - \sigma_{ss}}{\sigma_{tr} - \sigma_{ss}}\left(\log t_{rt} - \log t_{rs}\right) + \log t_{rs} & \sigma_{ss} < \sigma_\theta < \sigma_{tr} \\ \log t_{rt} & \sigma_\theta \ge \sigma_{tr} \end{cases} \tag{9.2-116}$$

$$\sigma_{ss} = 670.0 - 0.7(T - 273.15) \tag{9.2-117}$$

$$\sigma_{tr} = 1370.0 - 1.7(T - 273.15) \tag{9.2-118}$$

When the cladding temperature is between 973.15 K and 1042.15 K, the rupture time is determined using:

$$\log t_r = \begin{cases} \frac{T-973.15}{T_{ri}-973.15}(\log t_{rt} - \log t_{rs}) + \log t_{rs} & \sigma_\theta \le \sigma_{tr} \\ \log t_{rt} & \sigma_\theta \ge \sigma_{tr} \end{cases} \tag{9.2-119}$$

$$T_{ri} = \frac{2000 - \sigma_\theta}{2.6} + 273.15 \tag{9.2-120}$$

$$\sigma_{tr} = 2000 - 2.6(T - 273.15) \tag{9.2-121}$$

At temperatures above 1042.15 K, the rupture time is determined using:

$$t_r = t_{rt} \tag{9.2-122}$$

9.2.9.1.1 Steady-state CDF model

$$t_{rs} = min(t_{r1}, t_{r2}) \tag{9.2-123}$$

$$log(t_{r1}) = -32.49 + \frac{57781}{T} - \frac{11800}{T} log(\sigma_\theta) \tag{9.2-124}$$

$$log(t_{r2}) = -35.173 + \frac{45858}{T} - \frac{5563}{T} \log(\sigma_\theta) \tag{9.2-125}$$

where t_r is the clad rupture time (h), T is the clad Temperature (K), and σ_θ is the clad Hoop Stress (MPa) [9-30].

9.2.9.1.2 Transient CDF Model

$$dx0 = \tanh\left[2.0 \times 10^{-2} \times (\sigma_\theta - 200)\right] \tag{9.2-126}$$

$$dx1 = \tanh\left[\frac{\frac{dT}{dt} - 58}{17}\right] \tag{9.2-127}$$

$$dx2 = (-0.5 \times (1 + dx0)) \times (0.75 \times (1 + dx1)) \tag{9.2-128}$$

$$ax2 = -34.8 + dx0 + dx2 \tag{9.2-129}$$

$$bx2 = \frac{12}{1.5 + 0.5 \times dx0} \tag{9.2-130}$$

$$wx2 = ax2 + bx2 \times \log\left[\log\frac{\sigma^*}{\sigma_\theta}\right] \tag{9.2-131}$$

$$t_{rt} = e^{wx2} \times e^{\frac{Q}{RT}} \tag{9.2-132}$$

where $\sigma^* = 730$ (MPA), Q = 70107 (Cal/mol), R=1.987 (Cal/mol/K), σ_θ is the clad hoop stress (MPa), T is the Temperature (K), $\frac{dT}{dt}$ is the rate of change of clad temperature (K/s), and t_r is the clad rupture time (s), [Section 9.8.8.2.4].

9.2.9.2 SS316 and D9 Steady-state CDF Model

The SS316 and D9 pre-transient CDF model determines the rupture time based on the temperature and stress [9-33]. The time to rupture is determined using:

$$t_r = \theta(\sigma) \exp\left(\frac{Q}{RT}\right) f(D) \tag{9.2-133}$$

where σ is the stress (MPa), T is the temperature (K), $Q = 66$ is the activation energy kCal/mole, R is the gas constant (cal/mol-K), $\theta(\sigma)$ is the Dorn creep strength, obtained from [9-33], $D = D_c\psi_m$ is the cladding dose (dpa), ψ_m is the modified neutron fluence (#/cm^2/1.0E22), ψ is the neutron fluence (#/cm^2/1.0E22), $f(D)$ is a material-dependent irradiation induced embrittlement factor, and t_r is the clad rupture time (hr). The modeling of the irradiation induced embrittlement is derived from cladding void swelling data and the interrelation between void swelling and cladding embrittlement based on the creep rupture data given in Refs. [8.41], [8.42], and [8.43]. The irradiation induced embrittlement factor linearly interpolated from the data presented in Table 9.2.6 ψ_m is defined as

$$\psi_m = \begin{cases} \psi & \text{Clad = SS316} \\ 0.5\psi & \text{Clad = D9} \end{cases}$$

Table 9.2.6: Data points for SS316 and D9 embrittlement factor

SS316 Dose (dpa)	D9 Dose (dpa)	Value
0	0	1
30	50	1
50	80	0.3
55	90	0.1
60	100	0.01

9.2.9.3 SS316 and D9 Transient CDF Model

The SS316 and D9 transient CDF model determines the rupture time based on the temperature and stress [9-19]. The time to rupture is determined using:

$$T_r = -0.28 + 1.18 \tanh\left[0.5\left(1 - \ln\left(\frac{dT}{dt}\right)\right)\right] \tag{9.2-134}$$

$$d_0 = \tanh\left[\frac{1000 - T_{irr}}{200}\right] \tag{9.2-135}$$

$$d_1 = \tanh\left[\frac{\psi_m}{2}\right] \tag{9.2-136}$$

$$d_2 = \tanh\left[\left(\frac{\min(1,\sigma)}{550}\right)^{10}\right] \tag{9.2-137}$$

$$d_3 = 1 - \tanh\left[\left(\frac{T_{irr}}{583} - 0.438\right)^{25}\right] \tag{9.2-138}$$

$$d_4 = \tanh\left[\frac{\psi_m}{2.5}\right] \tag{9.2-139}$$

$$\sigma_s = 775 - 387.5\,(1 - d_0)\;\; d_1 + 125 d_2 d_3 d_4 \tag{9.2-140}$$

$$\ln(t_r) = -43.6 + 7.312 \log\left(\log\left(\frac{\sigma_s}{\sigma}\right)\right) - \frac{41399}{T} + T_r - 1.73 \tanh[\psi_m] \tag{9.2-141}$$

where σ is the stress (MPa), T is the temperature (K), $\frac{dT}{dt}$ is the rate of change of clad temperature (K/s), ψ_m is the modified neutron fluence (#/cm^2/1.0E22), T_{irr} is the temperature of cladding during irradiation (K), and t_r is the clad rupture time (hr).

9.2.9.4 Numerical Solution

It is assumed that at time = 0, the CDF is zero. For each channel and axial node, the CDF is computed using:

$$CDF^t = CDF^{t-1} + \frac{\Delta t}{t_r^t(\sigma, T)} \tag{9.2-142}$$

where Δt is the time step (s), t_r^t is the time to rupture at a given stress and temperature, σ is the stress (Pa), and T is the temperature (K).

9.2.10 Mechanistic Clad Creep Rupture

A mechanistic-based and semi-empirical clad failure model has been developed for HT9 cladding. The model aims at capturing the cavity dynamics; nucleation, growth, and coalescence behavior; cavity morphological evolution, and its connection with the rupture behavior. The interactions between cavities are modeled similar to a previously-developed mechanistic approach for fission gas bubbles [9-10], [9-11]. The cavity growth is modeled using the assumption and equation for the constrained diffusional cavity growth [9-20], [9-21]. Furthermore, the growth model is extended to account for the effect of the heterogeneities such as grain boundary carbides on the nucleation and growth behavior. A schematic description of the model is shown in Figure 9.2.3. Three different cavity types are allowed based on their sizes. The selected radius for each type is 0.05, 0.25, and 1.25 μm, respectively, based on experimental observations [9-20]. Small cavities nucleate at the grain boundaries proportional to the creep rate. The cavity growth under the applied stress occurs via creep and grain boundary diffusion. The growth rate of the cavities is used to compute the coalescence probabilities and the coalescence rate. As a result, the evolution of the cavity density and size is predicted. When the cavity areal fraction reaches 30%, failure is assumed.

The progression of the mechanistic creep failure (MCF) model is quantified using:

$$MCF = \frac{A_{cav}}{A_{cr}} \leq 1.0 \tag{9.2-143}$$

$$A_{cr} = 0.3\pi R_{gb}^2$$

$$A_{cav} = \pi\left(R_{c1}^2 N_{c1} + R_{c2}^2 N_{c2} + R_{c3}^2 N_{c3}\right)$$

where A_{cav} is the cavitated grain boundary area (m^2), A_{cr} is the critical cavitated grain boundary area(m^2), R_{gb} is the radius of the grain boundary, assumed to be 10 microns, N_{ci} is the number of type-i cavities, and R_{ci} is the radius of cavity type-i (m).

The number density of the different cavity types is determined using:

$$\frac{dN_{c1}}{dt} = J_{nucl} + J_{c1} - f_{12}\left(gc_{11} + gc_{12} + gc_{13}\right) - f_{13}\left(gc_{13} + gc_{31}\right) \tag{9.2-144}$$

$$\frac{dN_{c2}}{dt} = J_{c2} + f_{12}\left(gc_{11} + gc_{12} + gc_{13}\right) - f_{23}\left(gc_{22} + gc_{23} + gc_{32}\right) \tag{9.2-145}$$

$$\frac{dN_{b3}}{dt} = J_{c3} + f_{13}\left(gc_{13} + gc_{31}\right) + f_{23}\left(gc_{22} + gc_{23} + gc_{32}\right) \tag{9.2-146}$$

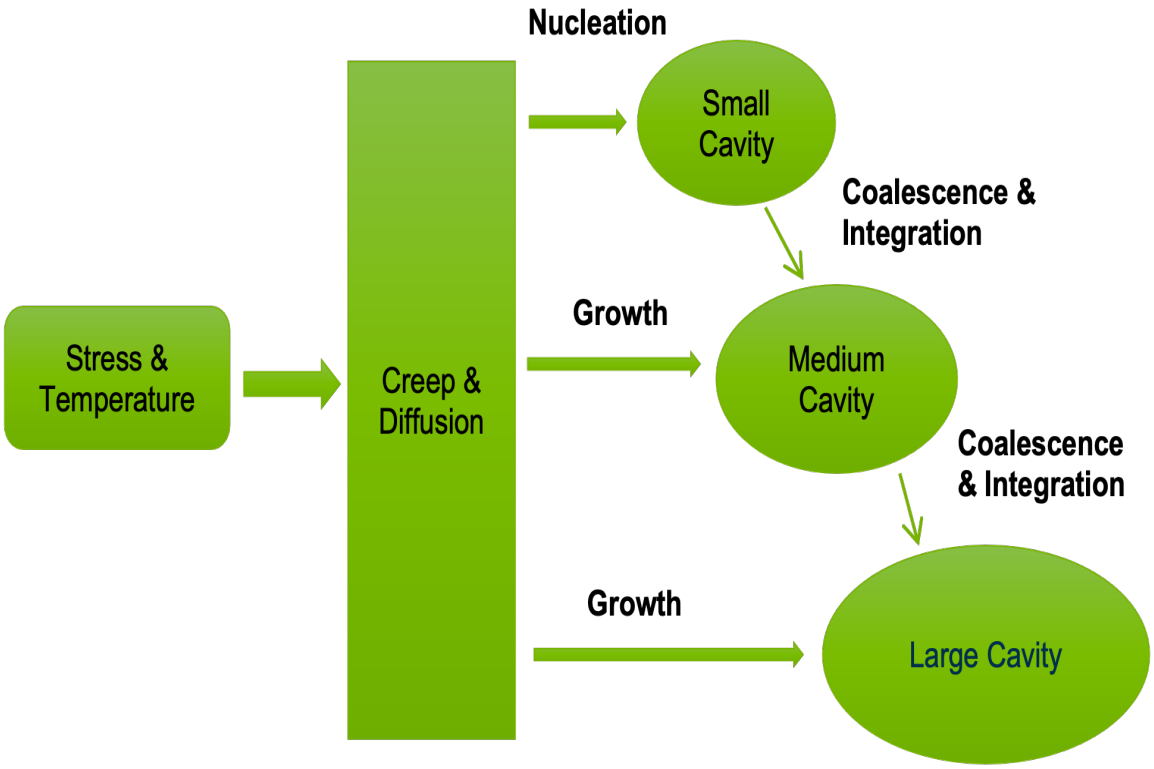


Figure 9.2.3: Schematic description of the mechanistic creep failure model (MCF)

where J_{ci} is the cavity surface density rate of change due to stress induced cavity growth (1/s), J_{nucl} is the nucleation rate of type-1 cavities $(1/s)$, gc_{ij} is the coalescence and integration rate of cavity-i into cavity-j due to growth of cavity-i $(1/s)$, and $f_{i,j}$ is the transition probability of cavity-i into cavity-j due to coalescences, $\frac{V_{cavi}}{V_{cavj}}$, V_{cavi} and V_{cavj} are the cavity volume of i and j cavity groups.

The MCF model is a system of ordinary differential equations and is solved using a backwards Euler method. The system of ordinary differential equations is solved independently for every axial node of the fuel pin. Prior to pre-transient irradiation, MFUEL assumes that MCF is zero.

9.2.10.1 Cavity Nucleation

Nucleation rate of cavities (J_{nuc}) is given as a function of a rate constant (K_{nuc}), creep rate ($\dot{\epsilon}_c$), and the grain boundary area (A_{gb}):

$$J_{nuc} = K_{nuc}\dot{\epsilon}_c A_{gb} \tag{9.2-147}$$

The nucleation constant depends on the operating stress, temperature, and microstructural heterogeneities. An empirical constant, which is called "G correction factor" has been derived in this work to account for the modeling imperfections associated with the nucleation and growth of the cavities and uncertainties in constitutive models. The nucleation constant is modeled as a function of the G correction factor. G is a tabular function depending on the stress and temperature as described in Section 9.8.6.

$$K_{nuc} = 1.0 \times 10^{12} G \tag{9.2-148}$$

9.2.10.2 Cavity Growth

Rice's constrained diffusional cavity growth model [9-20], [9-21] is adopted to compute the rate of change of the cavity size. However, the model is extended in several ways. First, continuous nucleation is modeled, allowing the spacing between the cavities and cavity density to change dynamically, in addition to change in the cavity size. Second, multiple sizes of cavity groups are allowed instead of single cavity size. These two improvements are much more consistent with the experimental observations. Finally, in order for the theoretical model to predict the real behavior including the microstructural complexities/heterogeneities, a correction factor, called G, is applied both to nucleation and growth rate. G correction factor is described in Section 9.8.6. The equation for the rate of change of cavity radius is given as follows:

$$\frac{dr}{dt} = G\frac{\sigma - (1-w)\sigma_0}{h(\Psi)r^2\left[\frac{q(w)kT}{2\Omega\delta D_b} + \frac{q'\sigma}{\dot{\epsilon}_c\lambda^2 d}\right]} \tag{9.2-149}$$

$$\sigma_0 = \frac{2\gamma_s sin\Psi}{2}$$

r: Cavity radius (m)

σ: Applied stress (Pa)

σ_0: Sintering stress (Pa)

d: Grain boundary facet =20 μm

λ: Effective cavity pitch (m)

w: $4r^2/\lambda^2$

Ω: Atomic volume = 1.18E-29 m^3

δ: Grain boundary thickness (m)

Ψ: Tip angle of the cavities $\cos\Psi = \frac{\gamma_b}{2\gamma_s}$

$$q(w) = -2\ln(w) - (3-w)(1-w)$$

$$h(w) = \frac{1}{sin\Psi}\left[\frac{1}{1+cos\Psi} - \frac{cos\Psi}{2}\right]$$

γ_b: Grain boundary free energy = 0.85 J/m^2

γ_s: Surface free energy = 2.1 J/m^2

G: Empirical constant to capture heterogeneity effects described in Section C.3.5

D_b: Grain boundary diffusion coefficient, $\delta D_b = 1.1 \times 10^{12} \exp\left(-\frac{1.75\times10^5}{RT}\right)$ (m^2/s)

R: Gas constant = 8.315 J/mol/K

$$q^{'} = \pi^2\left(1+\frac{3}{n}\right)^{1/2}$$

n: Power law stress exponent in creep relation = 6.6

9.2.10.3 Cavity Coalescence

The coalescence probabilities are calculated based on the derivations given in Ref. [9-11], [9-12]. The coalescence probability of cavity-i and cavity-j due to the growth of cavity-i, where i $\neq j$, is given as follows:

$$P_{ij} = \frac{\Delta r_i}{0.5l_j - r_i - r_j} \tag{9.2-150}$$

for i = j, the following relation is used:

$$P_{ij} = 2\frac{\Delta r_i}{l_j - r_i - r_j} \tag{9.2-151}$$

P_{ij}: Coalescence probability of cavity-i and cavity-j due to the growth of the cavity-i

Δr_i: Change in radius of cavity group-i at a given time step (m)

l_j: Distance between cavities belonging to group-j (m)

r_i: Cavity radius of group-I (m)

r_j: Cavity radius of group-j (m)

The resulting integration rate is the multiplication of the probability (P_{ij}) and smaller cavity number for i and j, N_{min}(i,j):

$$gc_{ij} = P_{ij}N_{\min(i,j)} \tag{9.2-152}$$

9.2.11 Clad Failure Assessment

Clad failure in metal fuel pins with ferritic/martensitic cladding is primarily thermal creep rupture that is augmented by the clad inner wastage formation. Thermal creep rupture can be assessed using thermal creep strain, CDF or MCF models. For a given SAS channel, when the thermal creep hoop strain reaches 1%, CDF or MCF reaches 1, the code prints a message indicating a clad failure prediction due to corresponding mechanism. While computing the clad stress, the model subtracts the clad wastage assuming the corresponding clad thickness does

not bear any load; however, when the clad wastage is excessively high, this approach may not be conservative. As a result, when 50% of the clad thickness turns into clad wastage, clad failure is assumed due to excessive clad wastage. The deterministic approach assumes that all fuel pins corresponding to the SAS channel failed.

These failure assessments cover the majority of the normal operation and transient scenarios. In addition, operation with excessive irradiation creep or void swelling strain (>2%) could lead to irradiation induced embrittlement, fuel assembly distortion, local undercooling and blockages. At current implementation, cladding failure does not activate the fuel failure relocation models.

9.3 Time Step Controls

The irradiation history is provided by the user for coarse time intervals. MFUEL subdivides these time intervals into approximately 10,000 s substeps and uses it as main time step. As described in Section 9.2, individual models may further subdivide the MFUEL's main time step. In transients, the maximum allowed time step in MFUEL is 0.01 s. If SAS's heat transfer time step is higher than 0.01 s, MFUEL will subdivide it to satisfy the time step requirement.

9.4 Solution Algorithm and Flow Charts

Figure 9.4.1 and Figure 9.4.2 show the MFUEL solution algorithm for pre-transient and transient analyses. MFUEL assumes the fuel pin can be modeled using a 1.5-dimensional approximation, therefore axial and radial loops are not explicitly shown in the algorithms. In each calculation step, MFUEL loops over each axial node independently and solves for the radial solution using a first-order approximation.

Pre-transient analysis is performed on a channel-by-channel basis during the SAS steady-state calculation. Based on user input, MFUEL solves for the changes to a fuel pin from the as-fabricated conditions to the start of the SAS transient simulation. As shown in Figure 9.4.1, the MFUEL pre-transient analysis loops over the user-defined irradiation history using substeps, as appropriate, to characterize the changes in the fuel pin. At the end of pre-transient characterization, MFUEL advances the fuel pin state to the match the user-defined SAS steady-state power and flow. With the new state, MFUEL updates the SAS axial and radial channel mesh and prints the steady-state solution.

For each substep of the pre-transient analysis, an average value for the user-provided power and flow is determined. The average power and flow is passed to the SAS thermal-hydraulic solver to determine fuel pin temperatures using the fuel pin composition and geometry at the start of the substep. Using the average power, the fuel pin burnup is incremented, along with the neutron fluence and dose to the cladding. Given the fuel temperature, redistribution of the fuel constituents is determined. If contact between the fuel and cladding has been predicted, iron and lanthanide migration is included. The coolant temperature is used to determine the clad and sodium corrosion interaction, which in addition to the iron and lanthanide migration, is used to determine the thickness of compromised, or wasted, cladding at the inner and outer surfaces. Fission gas production, migration, and release is simulated to determine the impact on fuel swelling. Given the state of swelling and the fuel pin temperatures, mechanical analysis is performed to determine the radial and axial elongation of the fuel pin and cladding. With the state of the fuel pin now known, the Cumulative Damage Fraction (CDF) and Mechanistic Clad Failure (MCF) of the clad are evaluated to determine the failure probability. The CDF and MCF value, along with the clad dose and wastage thickness are then checked to determine if warning or error messages need to be printed, providing users with an indication that a parameter has exceeded its validated range or clad failure has been predicted. Finally, the plenum pressure is updated based on the expansion of the fuel and cladding, the amount of fission gas released, and the relocation of in-pin sodium.

Similar to the MFUEL pre-transient analysis, the MFUEL transient analysis is performed on a channel-by-channel basis. For channels that utilize the MFUEL model, fuel pin conditions are updated at the end of each

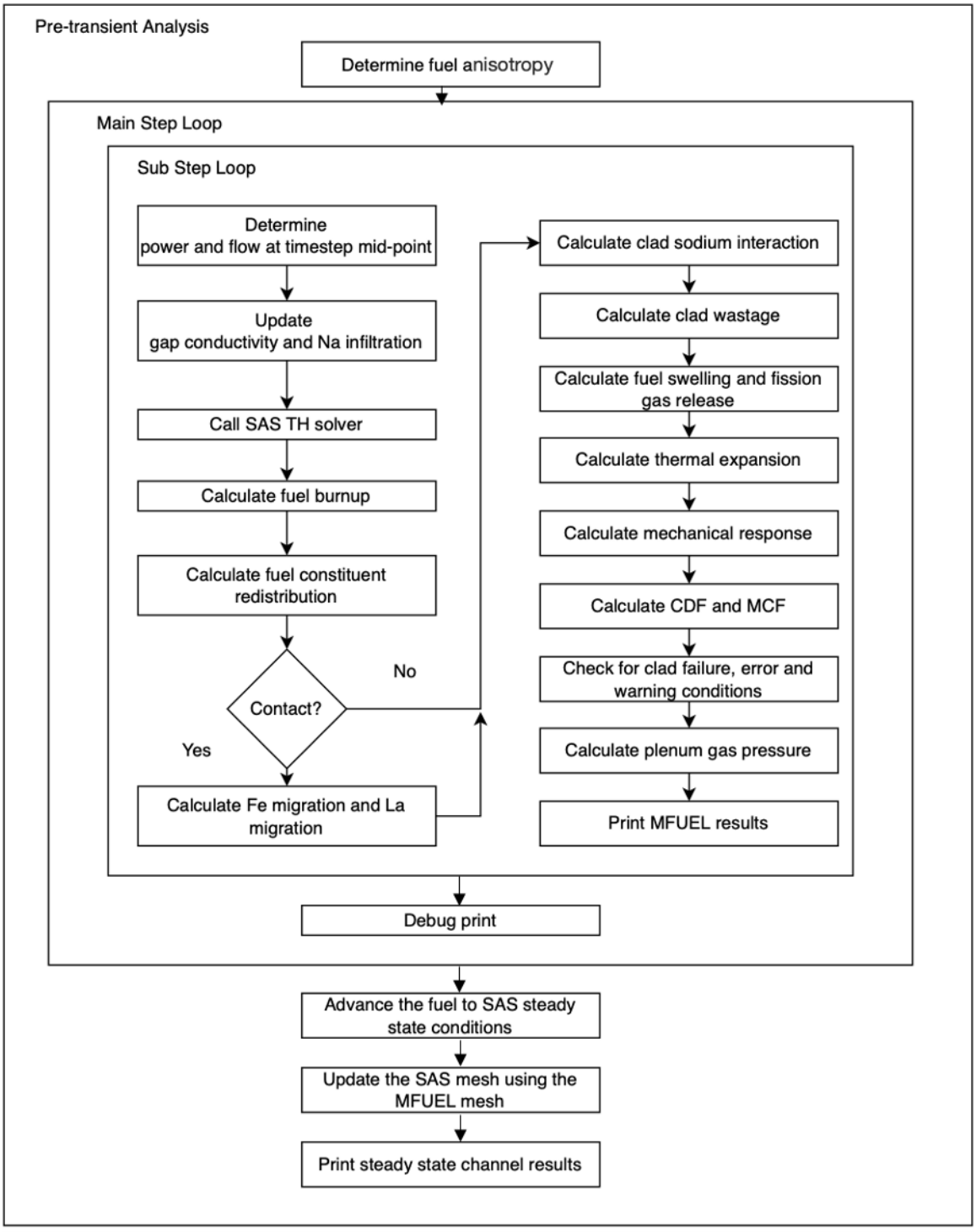


Figure 9.4.1: MFUEL Pre-transient Solution Algorithm

SAS heat transfer time step. The MFUEL transient analysis advances the state of the fuel pin using substeps, as shown in Figure 9.4.2. At the end of each main time step, MFUEL determines the reactivity contribution of the channel due to fuel and clad axial expansion.

For each sub-step of the transient analysis, the temperature and power of the fuel pin at the end of the main time step are used. SAS determines the temperature and power using the fuel pin geometry at the end of pre-transient and fuel pin composition at the start of main time step. Using the power, fuel pin burnup is incremented along with the neutron fluence and dose of the cladding. Given the fuel temperature, redistribution of the fuel constituents is determined. In the case of contact, lanthanide and iron migration are solved for. If the surface temperature of the fuel pin is high enough for eutectic formation to occur, iron migration is bypassed, and in its place the migration of the eutectic is determined. Corrosion due to sodium is tracked on the outside of the cladding. This internal and external wastage is used to determine the remaining thickness of the cladding for mechanical analyses. Fission gas production, migration, and release is simulated to determine the impact on fuel swelling. Given the state of swelling and the fuel pin temperature, mechanical analysis is performed to determine the radial and axial elongation of the fuel and cladding. With the state of the fuel pin now known, the CDF and MCF of the clad are evaluated to determine the failure probability. The CDF and MCF values, along with the clad dose and wastage thickness, are then checked to determine if warning or error messages need to be printed, providing users with an indication that a parameter has exceeded its validated range or clad failure has been predicted. Finally, the plenum pressure is updated based on the expansion of the fuel and cladding, the amount of fission gas released, and the relocation of in-pin sodium.

9.5 Interactions with Other SAS Models

The pre-transient characterization driver of MFUEL_SSInit is called from main steady-state driver routine of SAS (SSTHRM). For each fuel channel with MFUEL activated, MFUEL_SSInit is called only once. The time loop to simulate pre-transient history is handled by MFUEL. Steady state thermal-hydraulics routines are called from MFUEL at every coarse time step. At the end of pre-transient characterization, fuel pin, flow area and axial height are updated to be used for the thermalhydraulics and reactivity feedback models during the transient. During the transient, at each heat transfer time step, MFUEL_TsStep is called explicitly from transient driver of SAS (TSTHRM). At each time step, the fuel and clad reactivity feedback is computed due to axial height change of fuel and cladding computed by MFUEL, respectively. MFUEL calculation directly affects the fuel thermal conductivity, coolant flow area, and fuel and clad reactivity feedback.

Currently, coupling of MFUEL with coolant boiling (BOIL), pre-failure molten fuel in-pin fuel relocation (PINACLE), and post-failure fuel dispersal (LEVITATE) models has not been accomplished. Therefore, these models cannot be used if MFUEL is activated.

A new class structure is created to support MFUEL data storage and fuel performance simulations. Table 9.5.1 describes the high-level procedures within the MFUEL class. In order to reduce the complexity of the MFUEL class, additional classes are defined for the fuel clad chemical interaction and Zr redistribution models, described in Table 9.5.2 and Table 9.5.3, respectively.

The new MFUEL model interfaces with the existing SAS models by accessing the CHAN data structure and the common blocks. At a high level, the MFUEL model accesses the fuel pin temperature and geometry. Following pre-transient characterization, the MFUEL model updates the fuel pin geometry stored within the CHAN data structure. During transient analysis, SAS utilizes the updated fuel pin geometry and composition to solve for the fuel pin temperature. The fuel composition is stored within the MFUEL data structure and made accessible to the SAS material property routines.

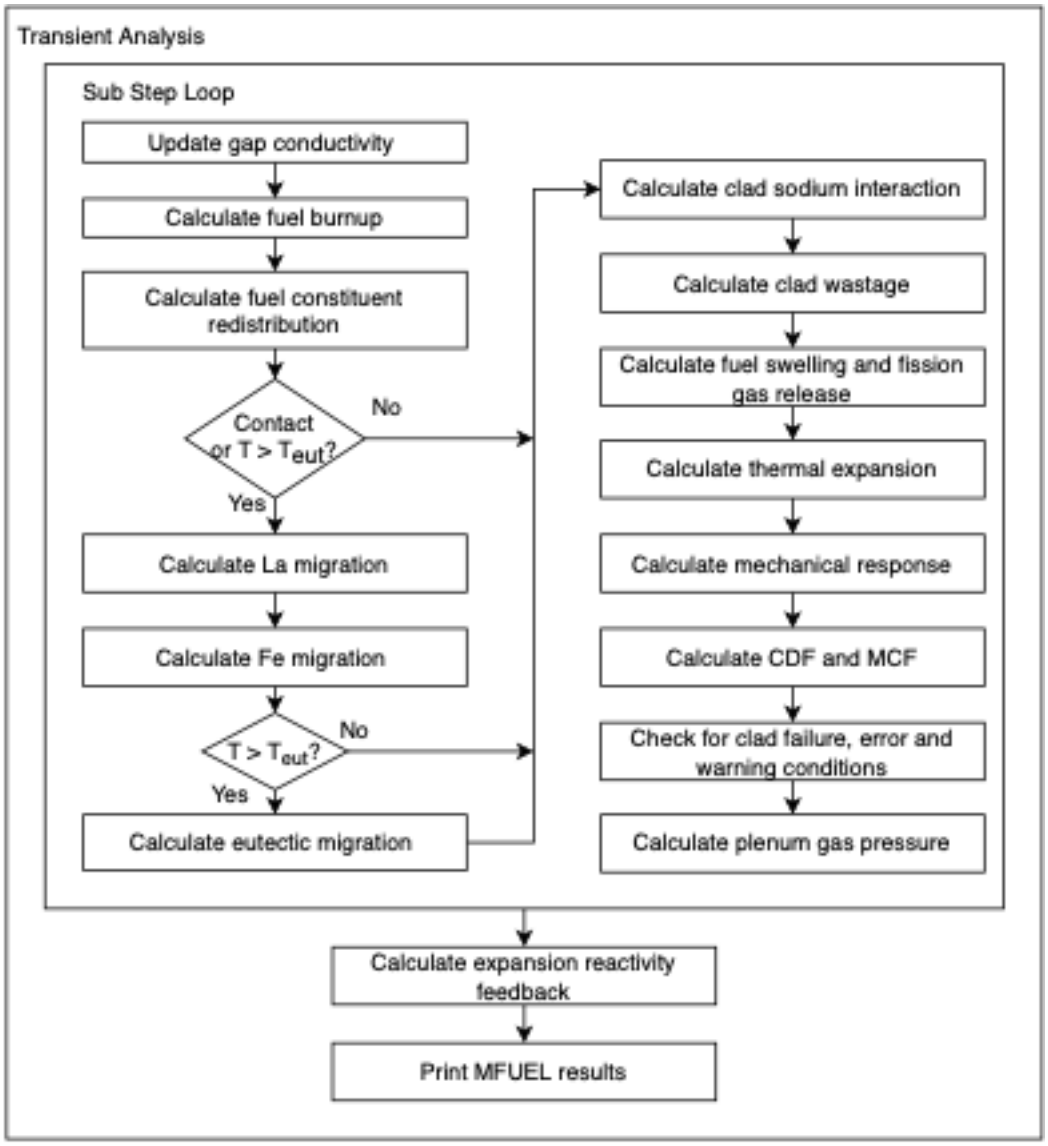


Figure 9.4.2: MFUEL Transient Solution Algorithm

Table 9.5.1: MFUEL Class Structure

MFUEL_BaseCH Method	Description
Init(ICH)	Initializes the memory and MFUEL model variables for channel ICH.
Release()	Releases the MFUEL data objects that were initialized during Init.
WriteDAT(unit)	Writes model specific restart data into unit.
ReadDAT(unit)	Reads model specific restart data from unit.
SSInit()	Simulates the pre-transient metal fuel behavior up to transient initialization. See Figure 9.4.1.
SSPrint()	Prints the MFUEL results and standard channel output at steady state.
TSStep()	Given a transient heat transfer time step, MFUEL computes the metal fuel behavior. See Figure 9.4.2.
TSPrint()	Prints MFUEL transient results.
FuelThermCond(i, j, T)	Computes the irradiated fuel thermal conductivity for node (i,j) at temperature T. Used if **IFUELM** = 3.
GetExpansionReactivity(rhoFuel, rhoClad)	Compute fuel, rhoFuel, and clad, rhoClad, reactivity change due to axial expansion or contraction.

Table 9.5.2: Fuel Clad Chemical Interaction (FCCI) Class Interface

FCCI Method	Description
Init()	Initializes FCCI model with a refined mesh relative to the SAS frame.
WriteDAT(unit)	Writes model specific restart data to unit.
ReadDAT(unit)	Reads model specific restart data from unit.
ActivateFCCI()	Activates fuel clad chemical interaction (actinide migration) within the LaMigration procedure and sets the boundary radial node between fuel and cladding.
UpdateGeometry()	Update FCCI mesh based on SAS frame. FCCI Mesh is always extended to the inner cladding wall.
UpdateTemperature()	Update FCCI temperatures based on temperatures in SAS frame.
LaMigration()	Computes the migration of lanthanides due to diffusion and clad inner wastage due to lanthanide precipitation.
FeMigration()	Computes the migration of iron, iron phase formation in the fuel and eutectic penetration into the cladding.
NaCorrosion()	Computes the clad outer wastage due to sodium corrosion.
GetFuelData(fData)	Transform data from the FCCI mesh to the SAS mesh.

Table 9.5.3: Fuel Constituent Redistribution (FCRD) Class Interface

FCRD Method	Description
Init()	Initializes FCRD model with a refined mesh relative to the SAS frame.
WriteDAT()	Writes model specific values calculated during steady-state initialization into RESTART.dat.
ReadDAT()	Reads model specific values from RESTART.bin.
UpdateGeometry()	Update FCRD mesh based on SAS frame.
UpdateTemperature()	Update FCRD temperatures based on temperatures in SAS frame.
ZrMigration()	Computes the migration of Zirconium due to concentration and temperature gradient driving force.
GetFuelData()	Get U and Zr atom fraction on the SAS frame.

Table 9.5.4: Fuel Swelling (SwellDriver) Class Interface

SwellDriver Method	Description
MFUEl_swell()	Computes fission gas and gas bubble behavior, swelling and release and solid fission product swelling.
MFUEL_creep()	Computes the creep induced hot pressing (pressure sintering) behavior of the fuel slug and irradiation/thermal creep, void swelling behavior of the cladding.
PoreYieldStress()	Computes the pore sintering yield stress.

Table 9.5.5: Fuel Pin Mechanical Analysis (StresDriver) Class Interface

StresDriver Method	Description
StresDriver()	Driver to call fuel pin mechanical analysis and update the fuel pin and coolant area geometry.
MFUEL_stress()	Given the strain components, it computes the fuel pin stress distribution and radial/axial displacements.
Contak()	Computes the contact stress between fuel and cladding.

Table 9.5.6: Other Modules

Method	Description
BurnupDriver()	Computes fuel burnup (at. fraction).
ExpanDriver()	Compute fuel and cladding incremental thermal expansion strain.
MCFDriver()	Computes mechanistic creep failure margin for the HT9 cladding only.
CLAD_props	Includes functions and subroutines for cladding irradiation, thermal creep, void swelling, thermal expansion, elastic constants, and CDF clad failure margin.
MFUEL_props	Fuel phase transition temperatures, empirical eutectic penetration model, and fuel elastic constants.
MFUEL_Parameters	Includes constant parameters for all models.
MFUEL_Data	Includes the derived data types for all models.

9.6 Input and Output Description

The user interface for the MFUEL model consists of text input that is part of the SAS4A/SASSYS-1 model file and text and binary output that is generated during a simulation. Existing input locations are used in conjunction with new input options to model the irradiation history, fuel specifications, and transient scenario.

Table 9.6.1 summarizes the input data that is utilized by the MFUEL model and provides valid input ranges as appropriate. Note that some of the existing input definitions have been updated such that they can be used by the MFUEL model. MFUEL can be activated by setting **IFUELO** = 2 for a given channel. In addition, **IFUELM** = 3 activates an alternative thermal conductivity correlation recommended for use with MFUEL. Since MFUEL simulates the fuel from the beginning of life, all fuel pin geometry parameters in SAS input should be set to as-fabricated conditions. Fuel porosity needs to be set to zero and initial plenum pressure needs to set to as-fabricated gas pressure at **TR**. Initial fuel composition should be provided as input. To model the irradiation history, the existing **MSTEP**, **PLIN**, and **TPLIN** inputs can be utilized, assuming that the SAS steady-state coolant flow rate is the flow rate during irradiation. Users can also use the existing TABLE block input format to define power and flow variations during pre-transient irradiation as a function of time. This option is utilized if **MSTEP** is set greater than 8, in which case **MSTEP** must correspond to a valid Table ID. A valid MFUEL table contains the columns TIME, POWER, and FLOW, APRINT, where the entries for TIME are defined in seconds, the entries for POWER are defined relative to POWTOT, and the entries for FLOW are defined in kg/s per pin, APRINT is used whether to skip (=1) or include (=0) printing of the axial distribution data to binary output files for the relevant MFUEL parameters.

Example irradiation history input using a *TABLE* block:

```
TABLE 101 "T473 Pre-Transient History"
        TIME            POWER           FLOW         APRINT
        0.00000E+00     2.78571E+00     4.95552E-02      0
        2.96165E+06     2.78571E+00     4.95552E-02      1
        2.96165E+06     2.73810E+00     5.06606E-02      0
        5.31213E+06     2.73810E+00     5.06606E-02      0
        5.31213E+06     2.83333E+00     5.06813E-02      0
        8.10774E+06     2.83333E+00     5.06813E-02      0
        8.10774E+06     2.85714E-01     4.81279E-02      0
        8.15276E+06     2.85714E-01     4.81279E-02      0
        8.15276E+06     2.85714E-01     4.81279E-02      1
        8.24237E+06     2.85714E-01     4.81279E-02      0
        8.24237E+06     1.96599E+00     5.03524E-02      0
        8.59148E+06     1.96599E+00     5.03524E-02      0
        8.59148E+06     3.33333E+00     6.50261E-02      0
        8.67860E+06     3.33333E+00     6.50261E-02      0
        8.67860E+06     2.76190E+00     4.99568E-02      1
        9.64807E+07     1.00000E+00     4.85789E-02      1
        9.65389E+07     1.00000E+00     4.85789E-02      1
END
```

Table 9.6.1: Input data utilized by the MFUEL model

Variable Name
ICLAD1
IPLUP
IYLD
KHDBK
YLDTAB

continues on next page

Table 9.6.1 – continued from previous page

Variable Name
YLDTEM
MSTEP
IRHOK
IMETAL
IFUELO
IFUELM
IEUTOPT
AXHI
IDSSC
PLENL
RBR
RER
RBRPL
RERPL
RINFP
ROUTFP
PLIN
TPLIN
FLTPOW
P0GAS
BONDNA
CDOSECONV
W0
RHOREF
WUREF
WPUREF
WZREF
FPRSTY
RAPIDEUT
HFMELT

MFUEL prints detailed steady state and transient results for the fuel characterization to standard SAS output. IF IDSSC input is set to 1, steady-state results will be printed for each user-controlled coarse time step. Otherwise, only end of steady state simulation results will be printed. Users has the ability to print results with a desired frequency for steady-state and transient simulations. Steady-state print frequency can be controlled by the size of the power/flow Table. Transient print frequency can be controlled by IPO and IPOBOI inputs. In addition to standard output, each SAS run with MFUEL activation creates binary files for each channel storing the peak values of essential parameters at every time step. Steady-state and transient binary files including the peak values of relevant parameters as a function of time are named as MFUELss_Ci.dat and MFUELts_Ci.dat, respectively where i is the channel number. Steady-state binary file including the axial distribution of the relevant parameters as time snapshots is named as MFUELss_Ci-axial.dat.

9.7 References

9-1. D. L. Porter, C. E. Lahm, R. G. Pahl, "Fuel Constituent Redistribution during the Early Stages of U-Pu-Zr Irradiation," Metallurgical Transactions A, Vol. 21, pp. 1871-1876, (1990).

9-2. R. G. Pahl, C. E. Lahm, S. L. Hayes, "Performance of HT9 Clad Metallic Fuel at High Temperature," Journal of Nuclear Materials, Vol. 204, pp. 141-147, (1993).

9-3. A. B. Cohen, H. Tsai, L. A. Neimark, "Fuel/cladding Compatibility in U-19Pu-10Zr/HT9-clad Fuel at Elevated Temperatures," Journal of Nuclear Materials, Vol. 204, pp. 244-251, (1993).

9-4. R. D. Mariani et al., "Lanthanides in Metallic Fuels: Their Behavior and Methods for Their Control," Journal of Nuclear Materials, Vol. 419, pp. 263-271, (2011).

9-5. K. Inagaki, T. Ogata, "Reaction of Lanthanide Elements with Fe-Cr Alloys," Journal of Nuclear Materials, Vol. 441, pp. 574-578, (2013).

9-6. K. Nakamura, T. Ogata, M. Kurata, A. Itoh, M. Akabori, "Reaction of U-Zr Alloy with Fe and Fe-Cr Alloy," Journal of Nuclear Materials, Vol. 275, pp. 246-254, (1999).

9-7. K. Nakamura, T. Ogata, M. Kurata, T. Yokoo, M. A. Mignanelli, "Reactions of Uranium-Plutonium Alloys with Iron," Journal of Nuclear Science and Technology, Vol. 38, No. 2, pp. 112-119, (2001).

9-8. T. H. Bauer, G. R. Fenske, J. M. Kramer, "Cladding Failure Margins for Metallic Fuel in the Integral Fast Reactor," SMIRT Intl. Conf. on Structural Mechanics in Reactor Tech., CONF-870812-22, Switzerland, (1987).

9-9. T. Ogata, T. Yokoo, "Development and Validation of ALFUS: An Irradiation Behavior Analysis Code for Metallic Fast Reactor Fuels," Nuclear Technology, Vol. 128, pp. 113-123, (1999).

9-10. C. B. Lee, D. H. Kim, Y. H. Jung, "Fission Gas Release and Swelling Model of Metallic Fast Reactor Fuel," Journal of Nuclear Materials, Vol. 288, pp.29-42, (2001).

9-11. A. Karahan, "Modeling of Thermo-Mechanical and Irradiation Behavior of Metallic and Oxide Fuels for Sodium Fast Reactors," Ph. D. Thesis, The Department of Nuclear Science and Engineering, Massachusetts Institute of Technology, Cambridge, MA, (2009).

9-12. A. Karahan, N. C. Andrews, "Extended Fuel Swelling Models and Ultra High Burn-up Fuel Behavior of U-Pu-Zr Metallic Fuel Using FEAST-METAL," Nuclear Engineering and Design, Vol. 258, pp. 26-34, (2013).

9-13. Y. Tsuoi, T. Ogata, M. Kinoshita, H. Saito, "Mechanistic Model of Fission Gas Behavior in Metallic Fuel," Journal of Nuclear Materials, Vol. 188, pp. 312-318, (1992).

9-14. D. R. Olander, "Fundamental Aspects of Nuclear Reactor Fuel Elements," Report # TID-26711-P1, United States of America, (1976).

9-15. G. L. HOFMAN et al., "Swelling Behavior of U-Pu-Zr Fuel," Metall. Trans. Vol 21A, p. 517, (1990).

9-16. T. H. Bauer, J. W. Holland, "In-Pile Measurement of Thermal Conductivity of Irradiated Metal Fuel," Nuclear Technology, Vol. 110, pp. 407-421, (1995).

9-17. R. P. Anantatmula et al. "Sodium Compatibility of HT-9 and Fe-9Cr-1Mo Steels," HEDL-SA-2801FP, (1985).

9-19. J. M. Kramer, R. J. DiMelfi, "An Evaluation of Transient Fuel Pin Cladding Failure Criteria for Application to Inherently Safe LMFBRs," ANL/RAS 84-6, Reactor Analysis and Safety Division, Argonne National Laboratory, (1984).

9-20. G. Eggeler, et al., "Microstructural Study of Creep Rupture in a 12% Chromium Ferritic Steel," Acta Metall., Vol. 37, No. 1, pp. 49-60, (1989).

9-21. J. R. Rice, “Constraints on the Diffusive Cavitation of Isolated Grain Boundary Facets in Creeping Polycrystals,” Acta Metall., Vol. 29, pp. 675-681, (1981).

9-22. G. L. Hofman et al., “Metallic Fuels Handbook,” ANL-NSE-3, Argonne National Laboratory, (2019).

9-23. L. Leibowitz, R. A. Blomquist, “Thermal Conductivity and Thermal Expansion of Stainless Steels D9 and HT9,” International Journal of Thermophysics, Vol. 9, No. 5, (1988).

9-24. S. M. McDeavitt, A. A. Solomon, “Hot-isostatic Pressing of U-10Zr by a coupled Grain Boundary Diffusion and Creep Cavitation Mechanism,” Journal of Nuclear Materials, Vol. 228, pp. 184-200, (1996).

9-25. T. H. Bauer et al., “First Overpower Tests of Metallic IFR Fuel in TREAT: Data and Analysis from Tests M5, M6, and M7,” ANL-IFR-124, Reactor Analysis and Safety Division, Argonne National Laboratory, (1989)

9-26. J. M. Kramer, Y. Y. Liu, M. C. Billone, and H. C. Tsai, “Modeling the Behavior of Metallic Fast Reactor Fuels during Extended Transients,” Journal of Nuclear Materials, Vol. 204, pp. 203-211, (1993).

9-27. Kim, Y. S, et al., “Modeling of Constituent Redistribution in U-Pu-Zr Metal Fuel,” Journal of Nuclear Materials, Vol. 359, pp. 17-28, (2006).

9-28. T. Kobayashi et al., “Development of the SESAME Metallic Fuel Performance Code,” Nuclear Technology, Vol. 89, pp. 183-193, (1990).

9-29. L. Leibowitz, R. A. Blomquist, “Thermal Conductivity and Thermal Expansion of Stainless Steels D9 and HT9,” International Journal of Thermophysics, Vol. 9, No. 5, (1988).

9-30. L. L. Briggs, et al., “Safety Analysis and Technical Basis for Establishing an Interim Burnup Limit for Mark-V and Mark-VA Fueled Subassemblies in EBR-II, ” ANL-NSE-1, Argonne National Laboratory, (2018).

9-31. B. J. Makenas, “Swelling of 316 Stainless Steel and D9 Cladding in FFTF,” CONF860605—45, (1986).

9-32. M. C. Billone et al., “Status of the Fuel Element modeling Codes for Metallic Fuels,” Proc. Int. Conf. Reliable Fuels for Liquid Metal Reactors, American Nuclear Society, Tucson, Arizona, (1986).

9-33. J. Lovell, B. A. Chin, E. R. Gilbert, “In-Reactor Creep Rupture of 20% cold-worked AISI316 Stainless Steel,” Journal of Materials Science, Issue 16, pp. 870-876, (1981)

9-34. D. Porter et al., “HT9 swelling in high burnup fast reactor fuel pin components,” Journal of Nuclear Materials, Issue 519, pp. 205-216, (2019)

9.8 Appendices

9.8.1 Appendix 9.1: Main Assumptions

The main assumptions associated with the MFUEL models are given as follows:

Fuel Constituent Redistribution

- Phase changes are assumed to occur rapidly compared with migration of alloy constituents.
- When a non-zero diffusion current is computed, U and Zr atoms switch places and Pu is immobile.
- Enthalpy of solution is negligible.
- Kinetic reactions terminate and equilibrium is assumed when a solubility limit is reached.
- Axial Zr migration is assumed to be negligible.

Fuel Clad Chemical Interaction

- Lanthanides can collectively be treated as diffusion in a single-phase medium due to their low solubility in metal fuel.
- Lanthanide diffusion into the cladding does not start until fuel-cladding soft contact.
- The brittle layer from lanthanide interaction does not bear any load.
- Eutectic formation of cladding is assumed to occur at any fuel cladding contact condition if the fuel surface temperature exceeds the eutectic temperature.
- The eutectic interaction layer in the cladding does not bear any load.
- Slow eutectic threshold temperatures vary as a function Pu content (e.g., 715 °C for U-10Zr) and is set to 1080 °C as constant for the rapid eutectic formation.

Fission Gas Behavior and Fuel Swelling

- Fission gas bubbles and interconnected porosity are modeled in six discrete groups.
- Bubbles are divided in to three groups based on constant atom number per bubble to model the observed bubble size distribution, bubble size evolution and their interactions.
- Bubbles nucleate uniformly in fuel matrix. It is assumed that phase boundaries are effective nucleation sites which are distributed uniformly.
- Gas swelling threshold for interconnected open porosity initiation is 10%.
- Fission yield of assumed to be 0.25 gas atoms/fission.
- Equilibrium bubble volume is assumed to behave based on the Van Der Waals EOS.
- The bubble shape correction factor is assumed to be different for spherical and ellipsoidal bubbles. $\alpha+\zeta+\delta$ phase forms ellipsoidal bubbles whereas single γ and $\beta+\zeta+\gamma$ phases form spherical bubbles.
- Nonequilibrium bubble behavior depends on the phases present, hydrostatic stress, bubble pressure, and fuel creep.

Plenum Pressure

- Plenum gas pressure is assumed to behave as an ideal gas assuming there is mechanical equilibrium between open porosity and plenum volume.
- Plenum gas volume accounts for the upper plenum and interconnected porosity in the fuel

Mechanical Analysis

- Plane strain is assumed for the computation of axial displacements, i.e., the area-averaged axial displacements are used to determine the axial displacement of a fuel axial node.
- Thin shell theory is assumed to compute cladding stresses. It is assumed that shear stresses are negligible in both the fuel and cladding.
- Fuel creep is assumed to be sufficiently high to avoid fuel stress gradients.
- All strain components are accounted for while computing the Fuel Clad Mechanical Interaction. Mostly fuel porosity sintering is the dominating balancing strain, which is assumed to be isotropic.
- When cracked fuel contacts with the cladding, it is assumed that axial growth is constrained by the cladding during the steady state.
- When the fuel is fully swollen, hard contact initiates such that fuel expansion can take place at the expense of porosity sintering which is assumed to be isotropic.
- Once eutectic forms between fuel and cladding, fuel clad mechanical interaction is assumed to be negligible. Note that at eutectic temperatures, fuel is fully single gamma phase which is very soft and porous.

- Upon eutectic formation, the default model assumes aggregated axial expansion without pore sintering consistent with experimental observation. Also a conservative option is provided for the users to allow for limited pore sintering in radial direction.
- Axial pressure due to fuel mass is accounted for.
- It is assumed that fuel swelling strain is isotropic. However, it is assumed that axial cracking in the fuel can lead to anisotropic fuel axial growth.
- It is assumed pore sintering/hot pressing occurs only to the open porosity bubble populations and is isotropic.
- It is assumed that cladding creep strain depends on the Von Mises or deviatoric stress.

In-Pin Sodium

- It is assumed that sodium does not infiltrate beyond 60% of the fuel radius, and only in the $\alpha + \zeta + \delta$ phase region. For fuel nodes beyond 60% of the fuel radius, it is assumed that sodium infiltration fills 30-60% of the available/open porosity, depending on the burnup level.

Corrosion due to Clad/Sodium Coolant Interactions

- It is assumed that sodium corrosion has a constant corrosion rate for a given temperature (no time/depth dependence).

Clad Failure Models

- In mechanistic clad creep rupture model, the grain boundary cavities are divided in to three groups based on their sizes to model the observed behavior, interaction between cavities, and growth/coalescence behavior.
- It is assumed that, for the mechanistic clad creep rupture models, when the cavitated grain boundary area reaches 30%, failure occurs.

9.8.2 Appendix 9.2: Thermodynamics and Kinetics Parameters of Fuel Constituent Redistribution Model

Diffusion constant and activation energy, heat of transport, and solubility limits for the phases present are given in Table 9.8.1, Table 9.8.2, Table 9.8.3, Table 9.8.4, respectively. Eq. (9.8-1) and Eq. (9.8-2) are used to predict the phase transition temperatures. Ref. [9-11] and [9-27] has been utilized to obtain most of the diffusion constants, diffusion activation energies and heat of transport values. Similar to Ref. [9-11] and [9-27], diffusion constants and heat of transport values are used as fitting parameters. Some of those parameters differs from the corresponding references. Phase transition Equations (Eq. (9.8-1) and Eq. (9.8-2)) are modeled as given in Ref. [9-11].

Table 9.8.1: Zirconium diffusion constant (m ^{2}/s)

Phase	Pu < 7 at%	Pu $\geq$ 7 at %
α	2.0e-07	4.0e-06
δ	2.0e-07	4.0e-06
β	5.7e-06	7.0e-04
γ	$0.4 \times 10^{-5.1-8.05C_{Zr}+9.13C_{Zr}^2}$	$10^{-5.1-8.05C_{Zr}+9.13C_{Zr}^2}$

Table 9.8.2: Zirconium activation energy Q_0 (kJ/mol)

Phase	Pu < 7 at%	Pu $\geq$ 7 at %
α	170.0	170.0
δ	150.0	150.0
β	180.0	180.0
γ	$128 - 107C_{Zr} + 174C_{Zr}^2$	$128 - 107C_{Zr} + 174C_{Zr}^2$

Table 9.8.3: Q_h - Heat of Transport (kJ/mol)

Phase	Pu < 7 at%	Pu $\geq$ 7 at %
α	0	150
δ	0	150
β	0	650
γ	-80	-200

Table 9.8.4: Zr Solubility Limit

Phase	Zr Solubility limit (at %)
α	5
δ	40
β	5

$$T_{\alpha+\delta\leftarrow\beta+\gamma} = \frac{a_{Pu}}{0.164}(868.15 - 935.15) + 935.15 \tag{9.8-1}$$

$$T_{\beta+\gamma\leftarrow\gamma} = \frac{a_{Pu}}{0.164}(923.15 - 965.15) + 965.15 \tag{9.8-2}$$

9.8.3 Appendix 9.3: Thermodynamics and Kinetics Parameters of Fuel Clad Chemical Interaction Model

Lanthanide Migration

The solubility limit of lanthanides in metallic fuel is assumed to be zero [9-4]. The lanthanide diffusion coefficient is given in Eq. (9.8-4), where S_{Dla} is the user defined sensitivity coefficient, **SCDLA**. Below 800 K, the temperature is set to 800 K in order to account for the irradiation-induced diffusivity. Ref. [9-5] is used to assess the solubility limit of lanthanides. Thermodynamic and kinetics parameters related the lanthanide diffusion and precipitation model are given in Table 9.8.5. Note that solubility limit of D9 cladding is reduced significantly compared to the HT9 cladding in order to account for excessive clad wastage behavior observed in experiments. Diffusion constant and diffusion activation energy is used as fitting factors.

$$D_{la} = \begin{cases} S_{Dla} D_0 exp\left(-\frac{Q_0}{RT}\right) & T \geq 750 \text{ K} \\ S_{Dla} D_0 exp\left(-\frac{Q_0}{750R}\right) & T \leq 750 \text{ K} \end{cases} \tag{9.8-3}$$

Table 9.8.5: Diffusion coefficient, heat of transport, and Lanthanide yield for Lanthanide migration

D_0 (m ^{2}/s)	10000.0
Q_0 (kJ/mol)	300.0
Y_{la} (#/J)	$15.29E+09$
Solubility in HT9 (atom/ m^3)	$6.4E+27$
Solubility in D9 (atom/ m^3)	$0.5E+27$
Solubility in SS316 (atom/ m^3)	$0.5E+27$
Solubility in 15-15Ti (atom/ m^3)	$0.5E+27$

Iron and Actinide Migration

The iron and actinide diffusion coefficients are calculated using the relationships given in Eq. (9.8-4) and Eq. (9.8-5), respectively, where S_{Dac} is the user defined sensitivity coefficient, **SCDAC**. Table 9.8.6 and Table 9.8.8 gives the diffusion constant and activation energy values. Table 9.8.9 and Table 9.8.10 gives the thermodynamics and kinetics parameters used in iron and actinide diffusion and eutectic formation. The experimental observartions in Ref. [9-6] is used to extract thermodynamics and kinetics parameters.

$$D_{Fe} = D_{0Fe} exp\left(-\frac{Q_{0Fe}}{RT}\right) \tag{9.8-4}$$

$$D_{Ac} = S_{Dac} D_{0Ac} exp\left(-\frac{Q_{0Ac}}{RT}\right) \tag{9.8-5}$$

Table 9.8.6: Diffusion constant and activation energy for iron diffusion in metal fuel

Parameter	T < $T_{\beta+\gamma}$	T ≥ $T_{\beta+\gamma}$
D_{0Fe} (m ^{2}/s)	0.225E-04	1.0E-04
Q_{0Fe} (kJ/mol)	220	220

Table 9.8.7: Diffusion constant and activation energy for iron diffusion in cladding

D_{0Fe} (m ^{2}/s)	8.5E-04
Q_{0Fe} (kJ/mol)	251

Table 9.8.8: Diffusion constant and activation energy for Actinides and eutectic

Parameter	T < 1074.15 K	1074.15 K ≤ T ≤ 1094.15 K	T ≥ 1094.15 K
D_{0Ac} (m ^{2}/s)	2.2E-08	$2.2E-08+3.3E-08\frac{T-1075.15}{20}$	5.5E-08
Q_{0Ac} (kJ/mol)	120	120	120

Table 9.8.9: Thermodynamics and kinetics parameters for iron, actinide diffusion, and eutectic formation

Phase Structure	Interaction Parameter (J/mol)	Iron Solubility Limit (%)	Phase growth activation energy (kJ/mol)
(U,Pu)Fe2	-3000	66.7	220
Fe2Zr + (U,Pu)6Fe	-1200	40.5	220
(U,Pu)6Fe + U-32Zr-50Fe	-750	28.6	220
(U,Pu)6Fe + U-42Zr-33Fe	-650	21.8	220
U-Zr matrix + U-42Zr-33Fe	-500	13.2	220
U-Zr matrix + U-23Zr-6Fe	-800	2.5	220
(U,Pu)Zr	-800	0.0	220
Clad	-600	0.05	220
Eutectic	-1600	0.5	220

Table 9.8.10: Phase growth constansts (1/s) for iron, actinide diffusion, and eutectic formation

Phase Structure	$T \leq T_{\beta+\gamma}$	$T \geq T_{\beta+\gamma}$
(U,Pu)Fe2	1.69E+05	7.5E+05
Fe2Zr + (U,Pu)6Fe	3.38E+05	1.5E+56
(U,Pu)6Fe + U-32Zr-50Fe	1.7E+06	7.5E+06
(U,Pu)6Fe + U-42Zr-33Fe	2.81E+06	1.25E+07
U-Zr matrix + U-42Zr-33Fe	3.94E+06	1.75E+07
U-Zr matrix + U-23Zr-6Fe	5.63E+07	1.5E+08
(U,Pu)Zr	0.0	0.0
Clad	0.0	0.0
Eutectic	2.81E+09	1.25E+10

9.8.4 Appendix 9.4: Fuel Swelling and Fission Gas Release Model Parameters

This section provides material properties and the constitutive models used in fuel swelling and fission gas release model. Table 9.8.11 and Table 9.8.12 summarize the parameters. Ref. [9-10] and Ref. [9-11] have been utilized to derive the constitutive models. Fission gas diffusion constants are the main fitting constants.

Fission Gas Diffusion Coefficient

The thermal bulk and surface fission gas diffusion coefficients, D_{fg_b} and D_{fg_s}, are dependent on the phase and irradiation conditions:

$$D_{fg_b} = \begin{cases} 10^{-4}\exp\left(-\frac{52{,}000}{RT}\right) + 10^{-4}\exp\left(-\frac{52{,}000}{RT_f}\right)F_r & T < T_\gamma \\ 10^{-8}\exp\left(-\frac{28{,}500}{RT}\right) + 10^{-4}\exp\left(-\frac{52{,}000}{RT_f}\right)F_r & else \end{cases} \tag{9.8-6}$$

$$D_{fg_s} = \begin{cases} 10^{-1}\exp\left(-\frac{52{,}000}{RT}\right) + 10^{-1}\exp\left(-\frac{52{,}000}{RT_f}\right)F_r & T < T_\gamma \\ 10^{-5}exp\left(-\frac{28{,}500}{RT}\right) + 10^{-1}\exp\left(-\frac{52{,}000}{RT_f}\right)F_r & else \end{cases} \tag{9.8-7}$$

where D_{fg_b} and D_{fg_s} are the bulk and surface fission gas diffusion coefficients (m^2/s), respectively, R is the gas constant, 1.987 cal/mol/K, T is the temperature in Kelvin, $T_f = 743.15$ is the temperature used to evaluate fission induced gas diffusion coefficient in Kelvin, and T_γ is the phase transition temperature, and F_r is the fission density rate (fission/m^3/s).

Bubble Nucleation

Bubble-1 nucleation rate is given as a function of matrix gas concentration and bubble-1 gas atom number as follows:

$$J_{b1,nucl} = k_{b1nuc} C_g \rho_{g1} \tag{9.8-8}$$

Where k_{b1nuc} is 10^{-20}(bub-1/s), C_g is the matrix gas atom density (atom/m^3), and ρ_{g1} is the constant atom number per bubble.

Gas Diffusion into the Bubbles

Diffusion of gas atoms into the bubbles can be calculated using the following equation:

$$J_{gi} = k_{gi} C_g N_{bi} A_{open} A_{ellipsoidal} \tag{9.8-9}$$

$$k_{gi} = E_{gbi} 4\pi r_{bi} D_g \tag{9.8-10}$$

where J_{gi} is the atomic flux into bubble-i by diffusion, k_{gi} is the gas diffusion constant to bubble-i (m^3/s), E_{gbi} is the empirical bias factor, currently set to 1, r_{bi} is the radius of bubble-i (m), D_g is the diffusion coefficient of gas atom (m^2/s), A_{open} is area correction for the open porosity network (the value is unity for the diffusion into closed bubbles), and $A_{ellipsoidal}$ is the area correction for the ellipsoidal bubbles (the value is unity for spherical bubbles and pores).

Integration rate via Bubble Diffusion

The integration rate of bubble-i into bubble-j due to collision between bubble-i and j is given as follows:

$$ab_{ij} = k_{ij} N_{bi} N_{bj} \rho_{gi} \tag{9.8-11}$$

$$k_{ij} = 4\pi \left(r_{bi} + r_{bj}\right)\left(D_{bi} + D_{bj}\right) \tag{9.8-12}$$

$$D_{bi} = \frac{3a_0^4}{2\pi r_{bi}^4} D_s \tag{9.8-13}$$

The integration rate of bubble-i into bubble-j due to collision between two bubble-i is given as follows:

$$ab_{ii} = 2k_{ii} N_{bi}^2 \rho_{gi} \tag{9.8-14}$$

$$k_{ii} = 16\pi r_{bi} D_{bi} \tag{9.8-15}$$

The transition probability of bubble-i into bubble-i+1, $f_{i,i+1}$, can be obtained by:

$$f_{i,i+1} = \frac{2\rho_{gi}}{\rho_{gi+1}} \tag{9.8-16}$$

where D_{bi} is the bubble-i diffusion coefficient (m^2/s), D_s is the surface diffusion coefficient (m^2/s), r_{bi} and r_{bj} are the radius of bubble-i and bubble-j (m), respectively, and $a_0^2 = 9 \times 10^{-20}$m^2.

Integration rate via Bubble Growth

$$gab_{ij} = P_{ij} \rho_{gi} N_{bi} \tag{9.8-17}$$

For collision between closed bubbles:

$$P_{ij} = \begin{cases} \frac{2\Delta r_{bi}}{0.5l_j - r_{bi} - r_{bi}} & i = j \\ \frac{\Delta r_{bi}}{0.5l_j - r_{bi} - r_{bj}} & i \neq j \end{cases} \tag{9.8-18}$$

For collision between closed bubble and open pores:

$$P_{ij} = \frac{\Delta r_{bi}}{l_j f_p} \tag{9.8-19}$$

Where P_{ij} is the probability of bubble-I colliding with bubble-j due to radial growth of bubble-i, Δr_{bi} is the radial growth of bubble-i with in a time step (m), r_{bi} and r_{bj} are the radius of bubble-i and bubble-j (m), respectively, l_j approximate distance between bubble-js, and f_p is an empirical fitting factor.

Table 9.8.11: Fission gas atom density ρ_g (atom)

Small Bubble	0.75E+08
Medium Bubble	0.2E+12
Large Bubble	0.25E+14

Table 9.8.12: Fuel swelling and fission gas release parameters

Y (atoms/fission)	0.25
k_{b1nuc} (bubble*atom/s)	1.0E-20
Threshold gas swelling (%)	10
A_{open}	0.1
$A_{ellipsoidal}$	1.4
f_p	0.7

9.8.5 Appendix 9.5: Mechanical Properties

The elastic and thermal expansion properties for metal fuel [9-22] [9-28], SS316, D9, HT9 and 15-15Ti cladding [8-17] [9-29], with temperature defined in Kelvin, are

Elastic Modulus (Pa)

$$E = \begin{cases} (56 + 0.1158\,(865 - T))\,10^9 & \text{phase} = \alpha + \delta \\ (56 + 0.2158\,(865 - T))\,10^9 & \text{phase} = \beta + \gamma \text{ or } \gamma \end{cases} \tag{9.8-20}$$

$$E = \begin{cases} E_c(T_f) \times 6.894757 \times 10^9 & \text{Clad} = \text{SS316 or 15-15Ti} \\ (2.137E + 05 - 102.74\,(T - 273.15))\,10^6 & \text{Clad} = \text{D9} \\ (2.137E + 05 - 102.74\,(T - 273.15))\,10^6 & \text{Clad} = \text{HT9} \end{cases} \tag{9.8-21}$$

where T is the temperature in Kelvin, T_f is the temperature in Fahrenheit, and $E_c(T_f)$ is defined in Eq. (8.7-22).

Poisson's Ratio (-)

$$\nu = \begin{cases} 0.3 & \text{phase} = \alpha + \delta \\ 0.3 & \text{phase} = \beta + \gamma \text{ or } \gamma \end{cases} \tag{9.8-22}$$

$$\nu = \begin{cases} 0.3 & \text{Clad} = \text{SS316} \\ 0.3 & \text{Clad} = \text{D9} \\ 0.3 & \text{Clad} = \text{HT9} \\ 0.3 & \text{Clad} = \text{15-15Ti} \end{cases} \tag{9.8-23}$$

Thermal Expansion Coefficient $(m/\Delta m)$

The composition-dependent thermal expansion coefficient for metallic fuel is determined by interpolating between alloy/component data in the Metallic Fuels Handbook [9-22].

The cladding thermal expansion strain is with temperature defined in Kelvin, and $T_0 = 293.15$ K, is

$$\alpha\left(T - T_0\right) = \begin{cases} \alpha_m(T_f)\left(T_f - 70\right)10^{-6} & \text{Clad = SS316 or 15-15Ti} \\ \left(-0.4247 + 1.282 \times 10^{-3}T\right. & \\ \left. +7.362 \times 10^{-7}T^2 - 2.069 \times 10^{-10}T^3\right)10^{-2} & \text{Clad = D9} \\ \left(-0.2191 + 5.678 \times 10^{-4}T\right. & \\ \left. +8.111 \times 10^{-7}T^2 - 2.576 \times 10^{-10}T^3\right)10^{-2} & \text{Clad = HT9} \end{cases} \tag{9.8-24}$$

where T is the temperature in Kelvin, T_f is the temperature in Fahrenheit, $T_0 = 293.15$ is a reference temperature in Kelvin, and $\alpha_m(T_f)$ is defined in Eq. (8.7-21).

Fuel Pore Sintering Yield Stress

Experimental data, although limited, shows evidence of pore sintering due to the softness of metallic fuel at elevated temperatures [9-24]. Prior to eutectic formation, any fuel expansion, caused by thermal expansion or fission product swelling, is balanced by pore sintering and fuel clad mechanical interaction. This balance is also evident in high-level experiments such as TREAT M-Series [9-25] and Whole Pin Furnace tests [9-26].

In order to account for the impact of pore sintering at elevated temperatures, a pore yield strength model, which is a function of creep rate and pore compressibility factor, is developed based on a reference data point in Ref. [9-24] and expert judgement. The selected reference point for pore yield strength is given in Table 9.8.13.

Table 9.8.13: Selected reference point for pore sintering yield stress

Reference Parameters	Reference Values
Temperature	973.15 K
Hydrostatic Stress	2.5 MPa
Pore Compressibility factor	C/6
C - Fitting Factor	6

Given temperature, hydrostatic stress, and fuel porosity, the model computes the pore compressibility factor (α_p), then solves the following equation to compute the pore yield strength(σ_f):

$$\epsilon_f(\sigma_f)\alpha_{pf} = \epsilon_{ref}(\sigma_{ref})\alpha_{pref} \tag{9.8-25}$$

where ϵ_f is the fuel equivalent creep rate (1/s) given the current temperature and hydrostatic stress, α_{pf} is the current fuel porosity compressibility factor (See Eq. (9.2-94)), ϵ_{ref} is the equivalent creep rate (1/s) computed using the parameters in Table 1, α_{pref} is the pore compressibility factor given in Table 1, and σ_f and σ_{ref} are the fuel pore yield strength (MPa) and reference stress (MPa), respectively. The assumed upper limits for pore yield strength are for porous fuel with more than 10% fuel porosity and low porosity fuel with less than 10% fuel porosity are 50 MPa and 80 MPa, respectively.

Yield Strength (Pa)

The cladding yield strength is defined in Section 8.7.10. The yield strength for SS316 is utilized for all four cladding types.

Ultimate Tensile Strength (Pa)

The cladding ultimate tensile strength is defined in Section 8.7.9. The ultimate tensile strength for SS316 is utilized for all four cladding types.

9.8.6 Appendix 9.6: Metallic Fuel Thermal Conductivity Model

The metallic fuel thermal conductivity equation given in Ref. [9-32] for the unirradiated U-Pu-Zr fuel is utilized. It is given as follows:

$$\begin{aligned} k_{0s} &= a + bT + cT^2 \\ a &= 17.5\left(\frac{1-2.23w_{Zr}}{1+1.61w_{Zr}} - 2.62w_{Pu}\right) \\ b &= 1.54\times10^{-2}\left(\frac{1+0.061w_{Zr}}{1+1.61W_{Zr}} + 0.9w_{Pu}\right) \\ c &= 9.38\times10^{-6}\left(1-2.7w_{Pu}\right) \end{aligned} \tag{9.8-26}$$

where k_{0s} is unirradiated solid metallic fuel thermal conductivity (W/m-K), T is temperature (K), a,b, and c are constants, w_{Pu}, w_{Zr} are the Plutonium and Zirconium weight fractions in fuel alloy

Irradiated Metal Fuel Thermal conductivity

Irradiated metallic fuel thermal conductivity is developed in this study in parallel with development and validation of MFUEL metallic fuel performance models of SAS.

$$k_{irr} = X_{Na}X_p k_0 \tag{9.8-27}$$

where k_{irr} effective thermal conductivity of irradiated fuel (W/m-K), X_{Na} is the thermal conductivity enhancement factor due to sodium infiltration into the fuel porosity, only applicable in outer $((\alpha+\zeta)+\delta)$ phase region Ref.[9-16], X_p is the thermal conductivity correction factor due to fission gas porosity, $P_{tot} = P_g + P_{Na}$, P_{tot}, P_g, P_{Na} are the total, gas filled, and sodium filled porosity.

$$X_p = (1-P_g)^{1.5} \tag{9.8-28}$$

$$X_{Na} = 1 - 3\frac{1-\left(\frac{k_{Na}}{0s}\right)}{1.163+1.837\left(\frac{k_{Na}}{k_{0s}}\right)}\frac{P_{Na}}{1-P_g} \tag{9.8-29}$$

where k_{Na} is the sodium thermal conductivity (W/m-K)

9.8.7 Appendix 9.7: MCF Model Parameters

Nucleation and growth constants (fitting factors) for Mechanistic Creep Failure (MCF) model is given in tabular form in Table 9.8.14.

Table 9.8.14: MCF model nucleation and growth fitting constants

Stress (MPa) Temp. (K)	50	75	100	125	150	200	250	300	350	500
773.15	-2.00	-2.25	-2.50	-2.40	-2.23	-2.00	-2.00	-2.00	-0.30	3.40
873.15	-2.00	-2.25	-2.50	-2.40	-2.10	-2.30	-0.10	1.15	1.80	1.95
973.15	-0.80	-0.55	-0.73	-0.65	-0.40	0.45	0.45	0.65	1.00	2.50
1073.15	0.65	0.92	-0.60	0.80	0.20	0.30	0.75	1.00	1.50	2.50
1173.15	0.10	0.70	-0.50	-0.10	0.00	1.00	1.00	1.00	1.50	3.00
1273.15	0.15	0.80	0.00	0.00	0.65	1.00	1.20	1.20	1.50	3.00
1373.15	0.10	1.75	0.20	0.00	0.70	1.20	1.20	1.20	1.50	3.00

9.8.8 Appendix 9.8: DEFORM-5: Metallic Fuel Cladding Transient Behavior Model

9.8.8.1 Introduction

The objective of the DEFORM-5 [9.8-1] model is to provide a predictive capability to quantify 1) margin to cladding failure, and 2) cladding failure timing and location in stainless steel-clad metallic fuel pins. It performs this function by using the current understanding of metallic fuel operating behavior and failure mechanisms in the form of mathematical formulations to model the relevant physical phenomena. (For a review of observed metallic fuel irradiation performance, see Ref. 9.8-2). In the DEFORM-5 model, calculated cladding mechanical response to transient thermal and pressure loading conditions is compared to expected failure characteristics and criteria to yield quantified measures of margin to failure, failure time, and failure location. DEFORM-5 applicability is focused on irradiated metallic fuel, in which the fuel has swollen into contact with the cladding.

9.8.8.2 Model Formulation

9.8.8.2.1 Fission Gas

Reference 9.8-2 reviews the current understanding of fission gas generation, release, and transport in metallic fuel. DEFORM-5 does not currently model steady-state fission gas generation and migration, but assumes that an independent assessment has been made to determine the fission gas content in the fuel element at the initial steady-state. The fission gas inventory is specified as input in the form of a axially uniform fuel element pressure at the SAS4A/SASSYS-1 initial reference temperature. The steady-state fission gas mass is then calculated from plenum and available pore volumes using the ideal gas law. Transient fuel element and internal fuel pressures are calculated using the calculated fission gas mass and the transient temperatures in the ideal gas formulation.

9.8.8.2.2 Fuel/Cladding Eutectic Alloy Formation

Uranium- and uranium/plutonium-based metallic fuels interact chemically with iron-based cladding to form low-melting-point eutectic alloys. For typical metallic fuel element geometry and for the transient sequences analyzed with SAS4A/SASSYS-1, the impact of this phenomena is to form a molten region at the fuel/cladding interface if the fuel is contacting the cladding, if the contact temperature has been raised to a sufficient level to cause eutectic alloy formation, and if the contact temperature has been held at the elevated level long enough to form the eutectic alloy, given the formation rate at that temperature. Figure 9.8.1 shows a binary phase diagram for uranium and iron [9.8-3]. This diagram shows that in equilibrium, liquid material may be formed at a temperature as low as 725°C for an alloy that is 89 wt. % uranium.

In SAS4A/SASSYS-1 analyses, where the fuel/cladding behavior model is concerned mainly with predicting margin to cladding failure, cladding failure time, and cladding failure location, the primary importance of eutectic alloy formation at the fuel cladding interface is an effective thinning of the cladding, and a lessening of its ability to contain the internal, hydrostatic pressure due to released fission gas. To quantify the cladding thinning in DEFORM-5, the correlation developed by Bauer [9.8-4] for the eutectic penetration rate as a function of absolute temperature is used. This correlation is

$$\dot{r} = 1 \times 10^{-6} \exp\left(22.847 - \frac{27624}{T}\right) \tag{9.8-30}$$

except in the range 1353 K (1080°C) to 1506 K (1233°C) where

$$\dot{r} = 1 \times 10^{-6} \left[922 + 2.9265\,(T - 1388) - 0.21522\,(T - 1388)^2 + 0.0011338\,(T - 1388)^3\right] \tag{9.8-31}$$

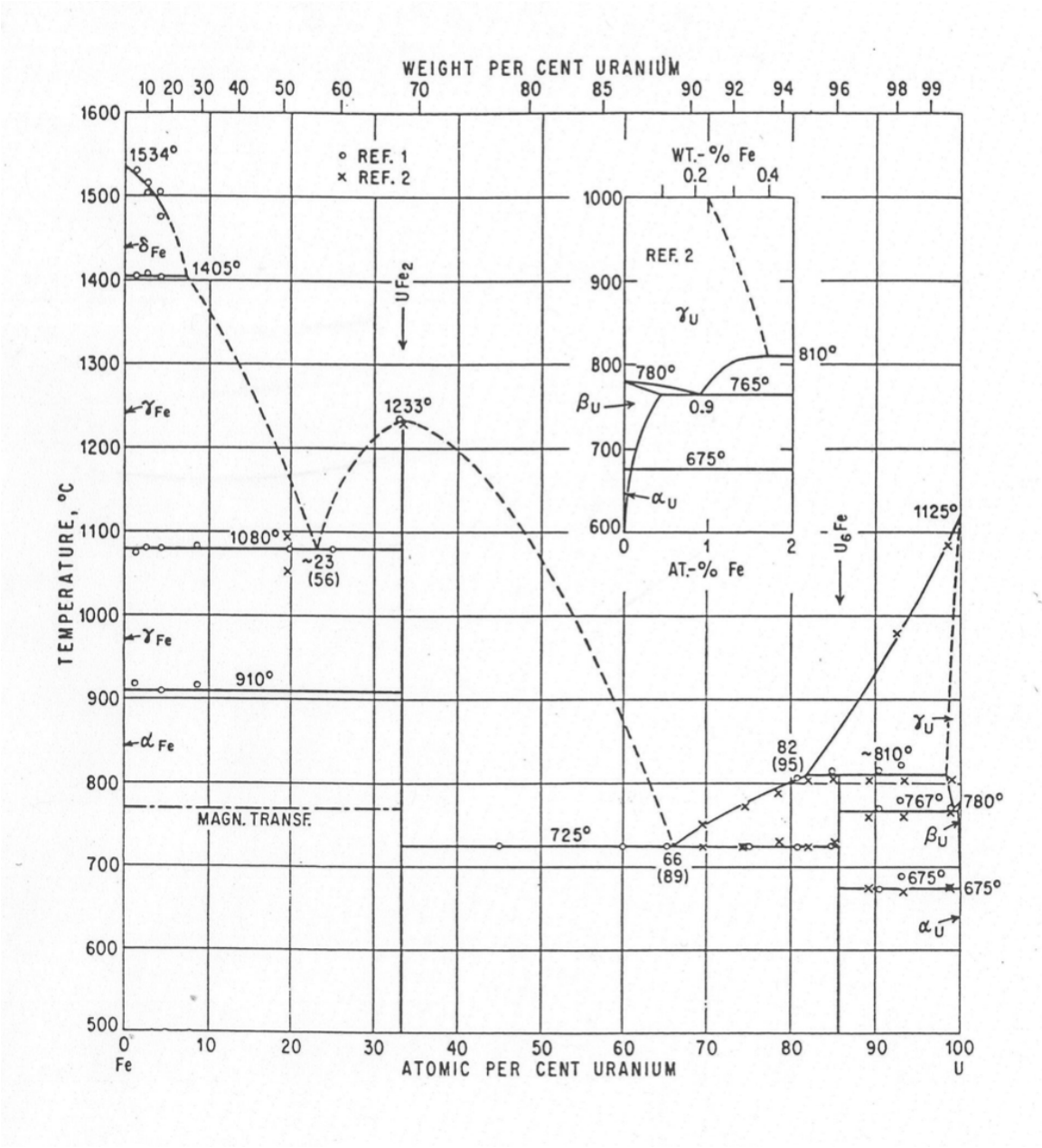


Figure 9.8.1: Uranium-Iron Phase Diagram [9.8-3].

Where $\dot{r}$ is the eutectic penetration rate in m/s, and T is the absolute temperature in Kelvins. This correlation is based on 1) tests in which iron capsules were dipped into molten uranium and uranium/iron alloy baths [9.8-5], 2) tests of EBR-II Mark-II driver fuel [9.8-6, 9.8-7, 9.8-8], and 3) tests of ternary alloy fuel (U-19Pu-10Zr) clad with stainless steel (D9) [9.8-4]. Waltar and Kelman [9.8-5] associated the rate increase in the range from 1353 K to 1506 K with the formation characteristics of the compound UFe_2. Eq. (9.8-30) and Eq. (9.8-31) are used in DEFORM-5 to calculate the cladding penetration as a function of time and the effective cladding thickness at each axial location and in each SAS4A/SASSYS-1 channel.

9.8.8.2.3 Clad Strain

In DEFORM-5, a calculation of the cladding strain is performed to provide input to the cladding failure assessment. The basic approach adopted for calculating cladding strain is that developed by DiMelfi and Kramer [9.8-9, 9.8-10] in their studies on plastic flow in steel cladding, and implemented in the FPIN2 computer code (see Section 11.3.6). The hoop stress in the cladding, σ_θ, is determined for a thin shell under internal pressure loading:

$$\sigma_\theta = [P - P_{\text{ch}}] \frac{a}{h} \tag{9.8-32}$$

where P is the internal pressure, P_{ch} is the coolant channel pressure, a is the inner cladding radius, and h is the cladding thickness. Assuming the hydrostatic loading of a thin, cylindrical shell, the hoop stress is converted to an equivalent stress:

$$\sigma = \frac{\sqrt{3}}{2}\sigma_\theta \tag{9.8-33}$$

and this stress is then used in the flow equation developed by DiMelfi and Kramer:

$$\dot{\epsilon} = \dot{\epsilon}_{\text{oos}} \left(\frac{\sigma}{\sigma_{\text{so}}}\right)^m \exp\left(\frac{-Q}{\text{RT}}\right) \tag{9.8-34}$$

where $\dot{\epsilon}$ is the equivalent strain rate, $\dot{\epsilon}_{\text{oos}}$ is a material constant, σ is the equivalent stress, σ_{so} is the saturation stress, m is the stress exponent, $\frac{Q}{R}$ is the creep activation temperature, and T is the absolute steel temperature.

This formulation was chosen because it allows the use of the same model for all cladding types of interest. Through the use of appropriate cladding parameters, SS316, D9, and HT-9 can all be modeled within the DEFORM-5 context. (In the absence of data, the following parameters for SS 316 are used for D9).

For type 316 stainless steel, Kramer and coworkers [9.8-11] derived the following parameters:

m = 5.35

$\frac{Q}{R} = 38,533$ K

$\dot{\epsilon}_{\text{oos}} = 1.062 \times 10^{14}$ s^{-1}

$\sigma_{\text{so}} = (2000 - 9.12T)\,(92,000 - 40.2T)$ Pa

For type HT-9 stainless steel, Kramer and coworkers [9.8-12] derived the following parameters:

$m = 2.263$

$\frac{Q}{R} = 36,739$ K

$\dot{\epsilon}_{\text{oos}} = 5.1966 \times 10^{10}$ s^{-1}

$\sigma_{\text{so}} = 3.956 \times 10^{-3} \times 2.12 \times 10^{11} \left(1.144 - 4.856 \times 10^{-4}T\right)$ Pa

9.8.8.2.4 Cladding Failure

The determination of cladding breach, or margin to cladding breach, is based on the eutectic thinning of the cladding and the reduced ability to contain the internal pressure. Besides the eutectic thinning, the cladding wall thickness is reduced as plastic flow takes place. Different accident scenarios and pin conditions lead to different modes of breach, whether through cladding penetration or plastic strain or a combination.

In the DEFORM-5 calculation, the time-dependent cladding damage fraction is calculated based on the time remaining to breach, following Kramer and DiMelfi's assessment [9.8-13] of the TDC-2 transient damage correlation for cold-worked 316 (and also for D9) stainless steel fuel pin cladding [9.8-14]:

$$\ln t_{\mathrm{r}} = A + B \ln \ln \left(\frac{\sigma^*}{\sigma} \right) + \frac{Q_1}{T} + \mathrm{TRAMP} + C \tanh \phi t \tag{9.8-35}$$

where

t_{r} = time to rupture, hours

A = -43.06

B = 7.312

C = -1.73

Q_1 = 41339

D = 1000

E = 200

F = 0.438

$\sigma^* = 775 - \left[387.5 - 387.5 \tanh\left(\frac{D-TI}{E}\right)\right] \tanh\left(\frac{\phi t}{2.0}\right)$
$+ 125 \left[\tanh\left(\frac{\sigma}{550}\right)^{10}\right] \left[1 - \tanh\left(\frac{\mathrm{TI}}{583} - F\right)^{25}\right] \tanh\left(\frac{\phi t}{2.5}\right)$

$\mathrm{TRAMP} = -0.28 + 1.18 \tanh\left(-0.5[\ln \dot{T} - 1]\right)$

σ = hoop stress, MPa (use thin wall formula with midwall tube diameter)

ϕt = fluence n/cm^2, E > 0.1 MeV, x10^{22}

T = transient temperature, K

TI = steady-state irradiation temperature, K

$\dot{T}$ = transient temperature ramp rate, K/sec

In DEFORM-5, the fast fluence is calculated from the input local linear power, the channel flux-to-power ratio, and number of full-power days of steady-state irradiation.

For HT-9, the time remaining to breach is assumed to be given by [9.8-15]:

$$t_{\mathrm{r}} = \theta \exp\left(\frac{Q}{RT}\right) \tag{9.8-36}$$

where

t_{r} = time to rupture, seconds

$\ln \theta = A + B \ln \ln \left(\frac{\sigma^*}{\sigma}\right)$

$A = -34.8 + \tanh\left(\frac{\sigma - 200}{50}\right) + C$

$B = \frac{12}{1.5+0.5\tanh\left(\frac{\sigma-200}{50}\right)}$

$C = -0.5\left[1+\tanh\left(\frac{\sigma-200}{50}\right)\right] 0.75\left[1+\tanh\left(\frac{\dot{T}-58}{17}\right)\right]$

σ^* = 730 MPa

σ = hoop stress, MPa

T = transient temperature, K

$\dot{T}$ = heating rate, K/s

Q = 70170 cal/mole

R = 1.986 cal/mole-K

9.8.8.3 Solution Techniques and Code Implementation

9.8.8.3.1 Initialization

The DEFORM-5 calculation is specified by setting ISSFUE = 0, ISSFU2 = 1, and IMETAL > 0 in the SAS4A/SASSYS-1 input deck. (See Table 9.8.15 for a listing of DEFORM-5-related input data). On the first transient time step, DEFORM-5 sets constants used in the strain rate and transient damage correlations and the eutectic penetration rate correlation, initializes integral program variables, and calculates the local cladding fluence as a function of the input-specified local linear power rating and input data **FLTPOW** and **FPDAYS**. If input variable **BURNFU** has been specified, then its numerical value is taken to be the cladding fluence in units of 10^{22} neutrons/cm^2; this is an approximation often used for EBR-II-irradiated fuel elements. If a steady-state irradiation temperature for use in cladding damage calculations is specified in input variable **TIRRFU**, it is stored on the first time step for future use. All of the first-time-step initialization is performed on entry to subroutine D5INIT.

If input variable **IPINFG** has the value 0, then on the first time step, the initial fuel element fill and fission gas masses for the DEFORM-5 fission gas model are calculated in subroutine FAILUR. Using the fission gas plenum geometry (See **PLENL** and **RBRPL**), the fuel geometry (See **AXHI** and **ROUTFP**), the fuel porosity (See **PRSTY**), an assumed fill pressure of 10^5 Pa, an input initial pressure (See **P0GAS**) at the reference temperature (See **TR**), the ideal gas constant (See **RGASSI**) and the molecular weights of the fission and fill gases (See **FGMM** and **HEMM**), the masses of fill gas and fission gas are computed from the ideal gas law.

On time steps following the first, entry D5INI2 is executed to save cladding temperatures, fission gas pressures, coolant pressures, and to calculate the time derivative of the cladding temperature at the beginning of the transient DEFORM-5 calculation.

Table 9.8.15: DEFORM-5 Input Data

Equation Symbol	Equation Number	Input	Block	Location	Suggested Value
-	-	IPO	1	12	>0
-	-	IPOBOI	1	13	>0
-	-	TR	13	419	27
-	-	FGMM	13	600	131
-	-	TFSOL	13	786	-
-	-	PRSTY	13	1073	-
-	-	RGASSI	13	1086	8.31434
-	-	HEMM	13	1225	4

continues on next page

Table 9.8.15 – continued from previous page

Equation Symbol	Equation Number	Input	Block	Location	Suggested Value
-	-	FIRLIM	13	1266	-
-	-	SECLIM	13	1267	-
-	-	THRLIM	13	1268	-
-	-	DTFAL1	13	1269	-
-	-	DTFAL2	13	1270	-
-	-	DTFAL3	13	1271	-
-	-	FGFI	13	1275	-
-	-	IFUELV	51	15	0
-	-	ISSFUE	51	32	0
-	-	MFAIL	51	86	-
-	-	ISSFU2	51	122	1
-	-	IMETAL	51	189	>0
-	-	IFUELC	51	193	0 or 1
-	-	ICTYPE	51	225	0
-	-	IGASRL	51	278	-
-	-	IRAPEN	51	280	0 or 1
-	-	IGSPRS	51	282	0 or 1
-	-	IFPIN2	51	285	0
-	-	IPINFG	51	486	0
-	-	IPORFG	51	488	-
-	-	AXHI	61	8	-
-	-	PLENL	61	53	-
-	-	RBRPL	61	102	-
-	-	ROUTFP	61	128	-
-	-	FPDAYS	62	1	-
-	-	FLTPOW	62	61	-
-	-	P0GAS	63	27	>0
-	-	FRUPT	64	81	-
-	-	FMELTM	65	2	-
-	-	BURNFU	65	54	-
-	-	TIRRFU	65	200	-

9.8.8.3.2 Fuel/Cladding Eutectic Alloy Formation

Eq. (9.8-30) is used to calculate the cladding penetration rate on the inner cladding surface, independent of the input-specified values of **IMETAL** and **ICTYPE**, in subroutine EUTPEN. The rapid penetration rate in the range from 1353 K to 1506 K is specified with a positive value of input variable **IRAPEN**; otherwise the rate is given by the exponential form in Eq. (9.8-30). In order for the rapid penetration rate to be effective, the local fuel surface temperature must be above the input fuel solidus temperature, **TFSOL**(**IFUELV**).

To accommodate large temperature changes on a heat transfer time step in fast transients, the temperature change across the time step is assumed to be linear and Eq. (9.8-30) is numerically integrated over the time step to avoid loss of accuracy. On each heat transfer time step, a new value of the cladding inner radius and the cladding thickness are calculated for use in subsequent cladding damage and strain rate calculations.

9.8.8.3.3 Fission Gas

On each time step, the fill and fission gas masses computed on the first time step, Section 9.8.8.3.1, are used with the transient temperatures to set the time-dependent internal-cladding and internal-fuel pressures. If input variable **IGSPRS** is set to 0, the pressure in the fuel porosity and the fission gas plenum is equilibrated in the transient to simulate rapid fission gas transport through the fuel, an option for long, slow accident sequences. Otherwise (IGSPRS > 0), steady-state gas in the fuel porosity remains in place and is heated with the fuel temperatures. In the transient, the volume of the fuel porosity is adjusted according to the input value of **IPORFG**, to estimate the impact of fuel density changes on porosity volume as the fuel changes temperature. These calculations are performed in subroutine FAILUR.

9.8.8.3.4 Cladding Strain

In the transient, subroutine STRANC computes the incremental cladding strains for each time step from the formulation in Section 9.8.8.2.3. If input variable **ICTYPE** has the value 1 or 2, then the parameters for 316 stainless steel are used. If ICTYPE has the value 3, then the parameters for HT-9 are used. The cladding inner radii and thicknesses at all axial locations are adjusted to reflect the transient cladding strains.

9.8.8.3.5 Cladding Failure

In the transient, subroutine FALMAR computes the cladding rupture time and the corresponding life fraction at all axial locations using the formulation in Section 9.8.8.2.4. The computed life fractions will be printed, but will not trigger cladding failure, and subsequent post-failure model execution, unless the input value of **MFAIL** has been set to 8 on input. In order to initiate LEVITATE or PLUTO2 with this failure criterion, it is also necessary to specify **FIRLIM**, **SECLIM**, **THRLIM**, **DTFAL1**, **DTFAL2**, **DTFAL3** and **FMELTM**.

In addition, the local coolant margin to boiling, expressed as the ratio of the absolute coolant temperature to the absolute saturation temperature, is also computed in FALMAR for subsequent printing.

Subroutine DFORM5 computes cladding failures to initiate plenum fission gas release in the coolant boiling model (See Chapter 12). For **IGASRL** > 0, input variable **FRUPT**(1) contains the cladding life fraction for failure of all the pins in the channel. This failure computation is independent of MFAIL.

9.8.8.3.6 Printed Output

DEFORM-5 printed output is produced from subroutine OUTPT5 on a main time step frequency set by input variables **IPO** and **IPOBOI**. When called, subroutine OUTPT5 prints the following quantities for all axial location: inner and outer cladding radii, cladding thickness, internal pin pressure, coolant channel pressure, cladding hoop stress and strain, incremental cladding strain and strain rate on the last step, cladding life fraction, coolant boiling fraction, time to rupture, time for cladding penetration, and the amount of penetration of the cladding.

9.8.8.4 References

9.8-1. K. J. Miles, "Metal Fuel Safety Performance," *Proceedings of the International Topical Meeting on Safety of Next Generation Power Reactors*, p. 119, American Nuclear Society, Seattle, Washington, May 1-5, 1988.

9.8-2. L. C. Walters, B. R. Seidel, and J. H. Kittel, "Performance of Metallic Fuels and Blankets in Liquid-Metal Fast Breeder Reactors," *Nucl. Tech.*, vol. 65, pp. 179-231, May 1984.

9.8-3. M. Hansen, "Constitution of Binary Alloys," McGraw-Hill Book Company, p. 728, New York, 1958.

9.8-4. T. H. Bauer, G. R. Fenske, and J. M. Kramer, "Cladding Failure Margins for Metallic Fuel in the Integral Fast Reactor," *Transactions of the 9th International Conference on Structural Mechanics in Reactor Technology*, vol. C, p. 31, Lausanne, Switzerland, August 17-21, 1987.

9.8-5. C. M. Walter and L. R. Kelman, "The Interaction of Iron with Molten Uranium," *J. Nucl. Matl.*, vol. 20, pp. 314-322, 1966.

9.8-6. P. R. Betten, J. H. Bottcher, and B. R. Seidel, "Eutectic-Penetration-Induced Cladding Rupture in EBR-II Driver Fuel Elements," *Trans. Am. Nucl. Soc.*, vol. 45, p. 300, 1983.

9.8-7. B. R. Seidel, "Metallic Fuel Cladding Eutectic Formation During Postirradiation Heating," *Trans. Am. Nucl. Soc.*, vol. 34, p. 210, 1980.

9.8-8. C. E. Lahm, et al., "EBR-II Driver Fuel Qualification for Loss-of-Flow and Loss-of-Heat-Sink Tests without Scram," *Nucl. Eng. Des.*, vol. 101, pp. 25-34, 1987.

9.8-9. R. J. DiMelfi and J. M. Kramer, "Modeling the Effects of Fast-Neutron Irradiation on the Subsequent Mechanical Behaviour of Type 316 Stainless Steel," *J. Nucl. Matl.*, vol. 89, pp. 338-346, 1980.

9.8-10. J. M. Kramer and R. J. DiMelfi, "Modeling Deformation and Failure of Fast Reactor Cladding During Simulated Accident Transients," *Nucl. Eng. Des.*, vol. 63, pp. 47-54, 1981.

9.8-11. J. M. Kramer, et al., Unpublished information, Argonne National Laboratory, 1978.

9.8-12. R. J. DiMelfi and J. M. Kramer, Unpublished information, Argonne National Laboratory, 1985.

9.8-13. J. M. Kramer and R. J. DiMelfi, Unpublished information, Argonne National Laboratory, 1984.

9.8-14. W. F. Brizes and M. L. Hamilton, Unpublished information, Hanford Engineering Development Laboratory, 1982.

9.8-15. M. L. Hamilton and N. S. Cannon, Unpublished information, Hanford Engineering Development Laboratory, 1985.

9.8.8.5 Nomenclature

Symbol	Description	Units
A	Variable or constant in TDC-2 correlation	-
B	Variable or constant in TDC-2 correlation	-
C	Variable or constant in TDC-2 correlation	-
D	Constant in TDC-2 correlation	-
E	Constant in TDC-2 correlation	-
F	Constant in TDC-2 correlation	-
m	Stress exponent	-
P	Internal fuel element pressure	Pa
P_{ch}	Coolant channel pressure	Pa
$\frac{Q}{R}$	Creep activation temperature	K
Q_1	Constant in TDC-2 correlation	-
$\dot{r}$	Penetration rate	m/s
t_{r}	Time to rupture in TDC-2 correlation	hrs,s
T	Cladding temperature	K
$\dot{T}$	Temperature rate of change	K/s
TI	Irradiation temperature in TDC-2 correlation	K
$\dot{\epsilon}$	Equivalent strain rate	1/s
$\dot{\epsilon}_{\text{oos}}$	Material constant	1/s
ϕ	Variable in TDC-2 correlation	
σ	Equivalent stress	Pa
σ	Hoop stress in TDC-2 correlation	MPa
σ_{so}	Saturation stress	Pa
σ_θ	Hoop stress	Pa
σ^*	Fitting coefficient in TDC-2 correlation	MPa
ϕt	Fast fluence in TDC-2 correlation	10^{22}n/cm^2

CHAPTER

TEN

SSCOMP: PRE-TRANSIENT CHARACTERIZATION OF METALLIC FUEL PINS

10.1 Introduction and Overview

The Integral Fast Reactor (IFR) Program [10-1] at Argonne National Laboratory is developing an advanced liquid-metal-cooled nuclear reactor (LMR) that uses U-Pu-Zr metallic alloy fuel due to a number of safety and economic benefits. The post-irradiation examination of U-Pu-Zr fuel pins irradiated in EBR-II shows the formation of three annular zones (central, middle and outer) of considerably different alloy compositions, fuel porosities and densities [10-2, 10-3]. The uranium in the fuel migrates from the central and outer zones to the middle zone, and the zirconium and fission products migrate in the opposite directions, i.e., from the middle zone to the central and outer zones [10-3]. Due to zirconium depletion the middle zone has a solidus temperature significantly lower than that of the central or outer zone. The zonal fuel densities have been measured to vary from about 8000 kg/m^3 in the central zone to about 16000 kg/m^3 in the middle zone [10-3]. The average porosity of each zone is also found to be significantly different from each other [10-4]. This radial redistribution of fuel constituents and annular zone formation has a considerable effect on the thermal behavior of U-Pu-Zr fuel pins because the thermal conductivity, specific heat and density of the fuel all vary with composition and porosity. The radial power shape within the fuel pin is also changed by fissile (i.e., U and Pu) redistribution.

During steady-state irradiation, the buildup of fission gas within the metal fuel produces considerably more fuel swelling than is observed in oxide fuel. As the steady-state irradiation proceeds in metal fuel, the fission gas is initially all retained in the fuel matrix. Grain boundary bubbles gradually form, producing fuel swelling. As the porosity continues to increase, the grain boundary bubbles can become interlinked, producing a long-range interlinked porosity that offers a path for fission gas release to pin plenum [10-4]. When metallic fuel was first used in EBR-II, a peak burnup of only 1 at. % was considered a reasonable objective because of the extensive fuel swelling caused by grain boundary bubbles. Once it was determined that this “breakaway” swelling appeared to be self-limiting, and if enough space (~25 to 30% of fabricated fuel volume) was provided in the fuel-cladding gap, the result was very little stress on the cladding and high burnups could be achieved [10-5]. In order to calculate the steady-state behavior of metal fuel, it is therefore necessary to model the fuel swelling due to grain boundary bubbles and the associated fission gas release in a consistent manner.

Listed below are the important in-pin phenomena (including the two described above) occurring during steady-state irradiation, that need to be modeled in order to characterize a U-Pu-Zr fuel pin at the beginning of a transient calculation:

1. Thermal expansion of fuel and cladding,
2. Fuel constituent radial migration,
3. Fission gas behavior, and porosity formation and distribution,
4. Irradiation-induced radial and axial swelling of fuel and cladding,

5. Bond sodium migration into fuel and pin plenum,
6. Cladding constituent migration into fuel.

The modeling of these phenomena in the SSCOMP module will provide SAS4A and SASSYS-1 codes with the capability to describe the transition of the fuel pin form cold clean as-manufactured conditions to hot irradiated swollen initial conditions for the transient fuel mechanics module FPIN2 [10-6]. The modeling of the following aspects of fission gas behavior (phenomenon 3 listed above) is required for calculating the initial conditions of FPIN2 transient calculation: (a) fission gas generation, (b) fission gas retained within grains, (c) fission gas retained in grain boundary bubbles that are not linked, (d) fission gas contained in grain boundary bubbles that are interlinked, and (e) fission gas released to pin plenum. The need for modeling these three types of fission gas contained in fuel (gas within grains, in unlinked grain boundary bubbles, and in interlinked grain boundary bubbles) has also been expressed by the pre-failure in-pin fuel motion module PINACLE development [10-7]. Regarding irradiation induced fuel swelling (phenomenon 4 listed above), axial swelling will contribute to pre-transient fuel column length, a needed initial condition for the FPIN2 module. Radial swelling of fuel and cladding will determine fuel-cladding gap closure and bond sodium migration into pin plenum, a needed initial condition for the FPIN2 module. The modeling of fuel swelling phenomenon has two important aspects: swelling due to solid fission products, and fission gas-induced swelling. Swelling of a cladding material is mainly determined by neutron fluence which is also a parameter in determining cladding rupture life used in the FPIN2 transient fuel mechanics calculation.

The SSCOMP module is being developed to model the above listed in-pin phenomena during steady-state irradiation of metallic fuel pins. After the fuel pin has been characterized at the end of steady-state irradiation, the calculation of the relevant ones of these phenomena during the transient will be performed by the DEFORM-5 model. The purpose of this chapter is to present the current status and future modeling needs of the SSCOMP module. As discussed in Section 10.2, the zone formation calculation method of the earlier SSCOMP model [10-4] is not incorporated into SAS4A/SASSYS-1 Version 3.0 code due to its limitations. The study of fuel constituent radial migration leading to zone formation was not at that time developed to a state where an appropriate dynamic model could be developed. At present, the data needed to account for the effects of zone formation on pin transient temperature calculation are specified through input data controlled by the input parameter IFUELC. As a starting point for SSCOMP development, Section 10.3 describes thermal properties of U-Pu-Zr metallic alloy fuel as a function of composition and temperature. These properties are needed in the analysis of the Mark-V fuel (U-20Pu-10Zr), and other metallic alloy fuels of different compositions, especially the different fuel compositions across the pin radius resulting from fuel constituent migration during steady-state irradiation. The implementation of the thermal properties data of the IFR Metallic Fuels Handbook [10-8], using a method of regionwise interpolation, is an important part of Section 10.3. Thermal properties of mixed-oxide fuel are described in Section 3.15, and not here. Future directions for modeling efforts are discussed in Section 10.4.

10.2 The Earlier SSCOMP Model and Current Capability

Assuming chemical equilibrium at annular zone boundaries and mass conservation of fuel constituents, a model was developed earlier [10-4] to directly calculate zone formation at the end of steady-state irradiation. This model was also calibrated using zone formation data obtained for the cross section at the mid-height of a U-Pu-Zr alloy fuel pin in an EBR-II steady-state irradiation experiment [10-3]. This earlier SSCOMP model calculates the alloy compositions of the zones, the boundary radii of the zones, and the resulting in-pin power radial profile. The earlier model does not calculate the time-development of the fuel zones from the fabricated condition to the end-of-steady-state irradiation condition. This is so because the study of fuel constituent radial migration leading to zone formation was not at that time developed to a state where an appropriate dynamic model could be developed.

The zone formation calculation method of the earlier SSCOMP model is not incorporated into SAS4A/SASSYS-1 Version 3.x codes due to its limitations. Some of its limitations are:

1. The earlier SSCOMP model directly calculates the fuel compositions and boundary radii of the zones based on chemical equilibrium under end-of-steady-state reactor operating conditions, and does not calculate the migration of fuel constituents with time during steady-state irradiation. Reactor operating conditions usually change during the steady-state irradiation of a fuel pin, and therefore, chemical equilibrium under end-of-steady-state operating conditions may not have been attained if the operating conditions changed during a (fuel constituent) migration time constant before the end of steady-state irradiation.
2. The earlier SSCOMP does not model other important pre-transient phenomena such as irradiation-induced fuel swelling, fission gas generation, retention and release to pin plenum, etc. These phenomena are important for characterizing a fuel pin at the beginning of a transient as noted in Section 10.1.
3. The earlier SSCOMP model is not well validated by analysis of experimental data for zone formation.

Therefore, after evaluating the earlier SSCOMP in light of new data on zone formation and current understanding of phenomena causing it (zone formation), the earlier model or another more advanced model will be incorporated into SAS4A/SASSYS-1 codes, and this will be done in the framework of models for other important pre-transient phenomena that characterize the fuel pin at the beginning of transient calculation. At present, the zone formation data (such as zonal U-Pu-Zr alloy compositions, boundary radii of the zones, etc.) are supplied to the code as input data, and used in pin transient temperature calculation. This multiple radial fuel zone option is controlled by the input parameter IFUELC.

10.3 Thermal Properties of U-Pu-Zr and U-Zr Alloy Fuels

The fuel thermal properties described in this section are used in all modules of SAS4A/SASSYS-1, and not only in the SSCOMP module. The fuel thermal properties include its specific heat, theoretical density and thermal conductivity. A U-Zr binary alloy is treated as a special case of the U-Pu-Zr ternary alloy. In the SAS4A/SASSYS-1 codes, one of the following four options are used for providing fuel thermal properties of a channel: (1) temperature-dependent property tables prepared by the user, and supplied to the code as input data, (2) regionwise interpolation of IFR Metallic Fuels Handbook data (as explained in Section 10.3.2, region here implies a triangular part of the U-Pu-Zr ternary system composition triangle), (3) composition- and temperature-dependent fuel property correlations of the report ANL/RAS 85-19 developed before IFR Metallic Fuels Handbook [10-9] in course of the earlier SSCOMP model development [10-4], and (4) Billone's correlations for Mark-V (i.e., U-20Pu-10Zr) and U-10Zr fuels based on IFR Handbook data. The options available for providing mixed-oxide fuel properties to the codes are described in Section 3.15, and not in the present section.

The specific heat of both metallic and oxide fuels is calculated in subroutine CFUEL, the theoretical density in subroutine RHOF, and the thermal conductivity in subroutines FK and KFUEL. The difference between the two thermal conductivity subroutines FK and KFUEL is that FK is used to calculate the thermal conductivity of a single fuel radial node during steady-state heat transfer calculation, whereas KFUEL is used to calculate the thermal conductivities of all fuel radial modes in a pin axial segment (in a single call, for computational efficiency) during transient heat transfer calculation. The flow diagrams of these subroutines are shown in Figure 10.3.1 to Figure 10.3.5. The above mentioned four options for providing fuel thermal properties along with some other options related to fuel properties, are controlled through the following six input parameters: IRHOK, IMETAL, IFUELM, IFUELC, KFIRR and KDENBU. The input parameter IMETAL>1 implies U-Pu-Zr alloy fuel.

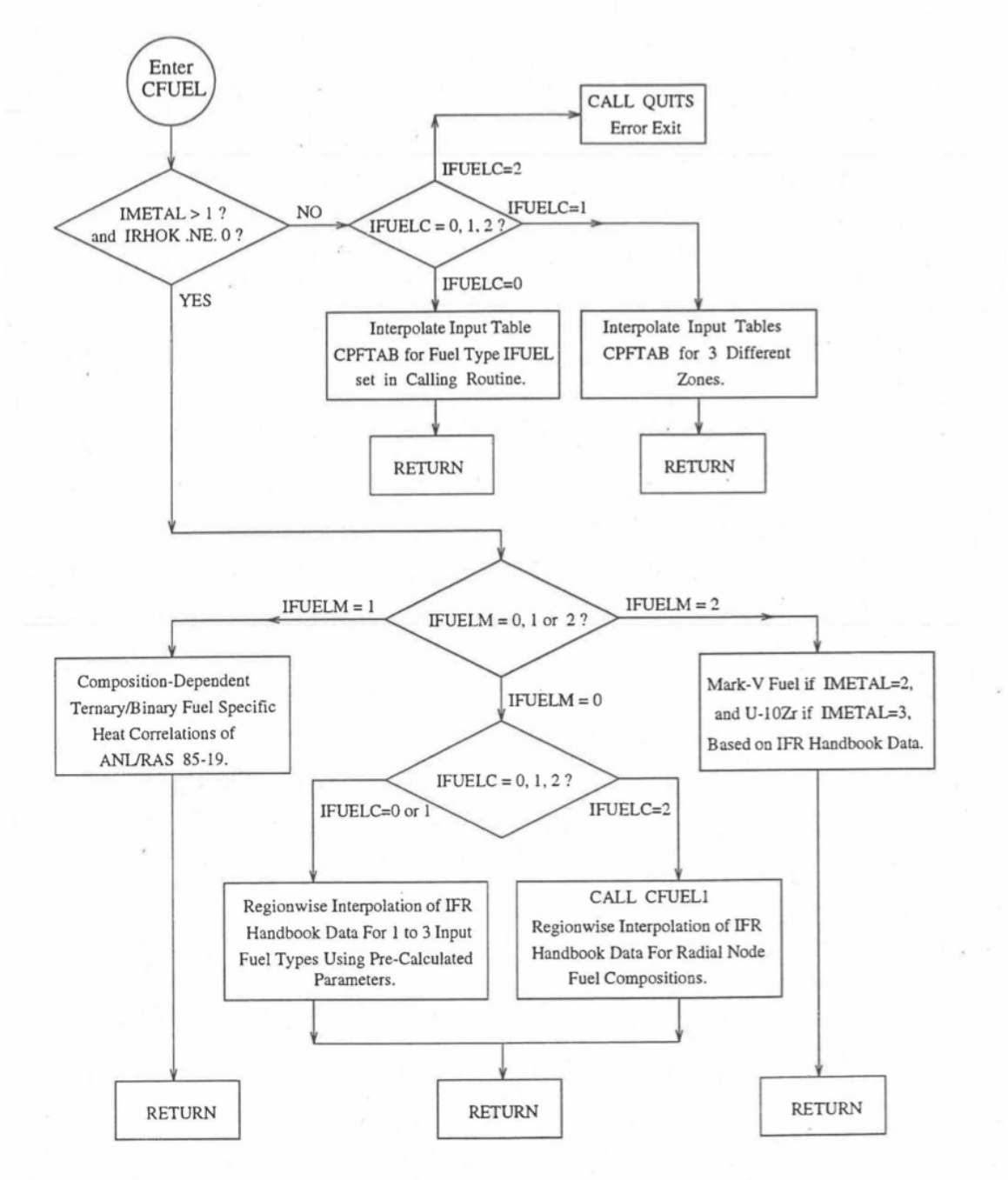


Figure 10.3.1: Flow Diagram of Subroutine CFUEL for Fuel Specific Heat.

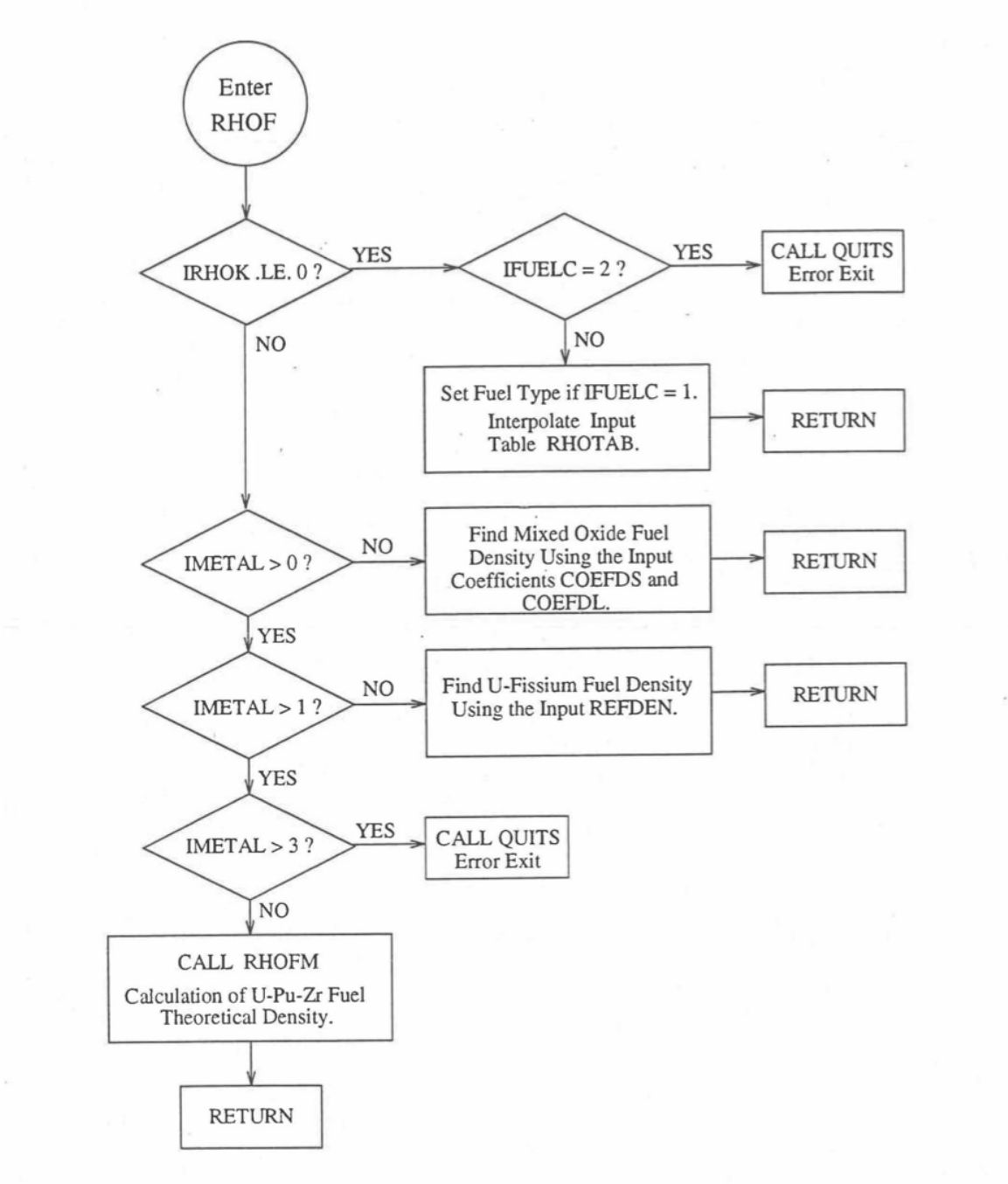


Figure 10.3.2: Flow Diagram of Function Subprogram RHOF for Fuel Theoretical Density.

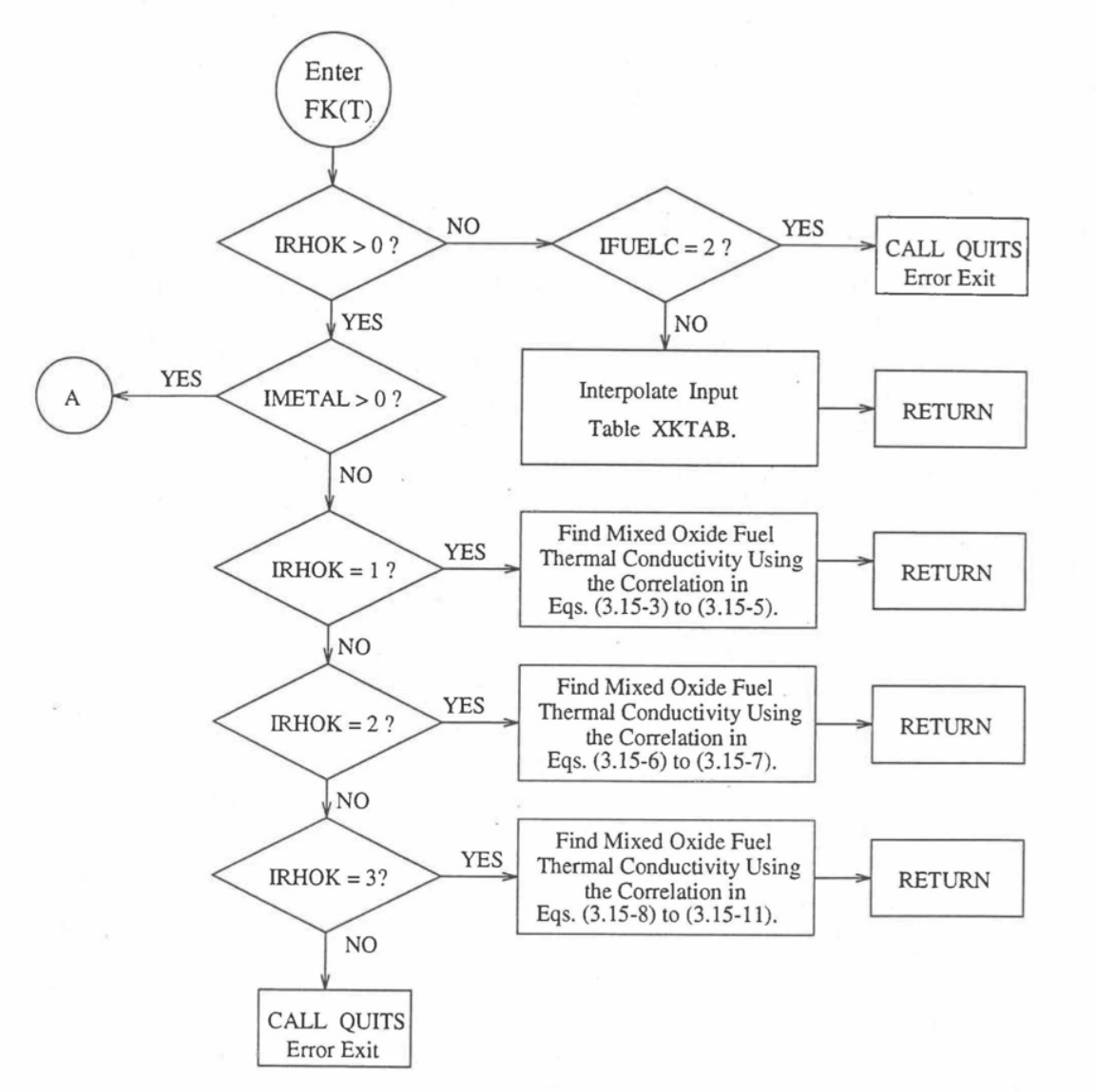


Figure 10.3.3: Flow Diagram of Function subprogram FK for Thermal Conductivity of a Single Fuel Radial Node.

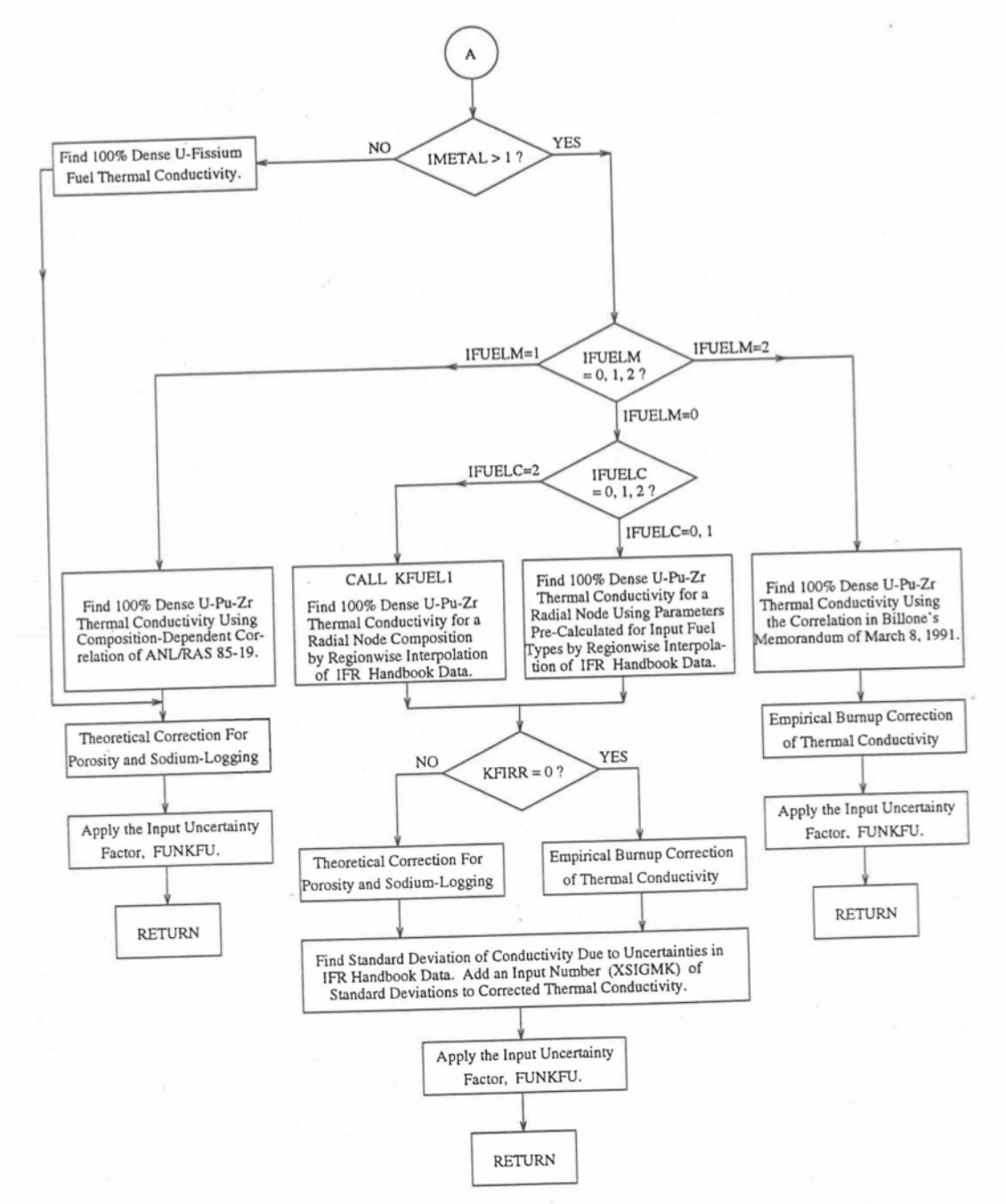


Figure 10.3.4: Flow Diagram of Function subprogram FK for Thermal Conductivity of a Single Fuel Radial Node (Cont'd).

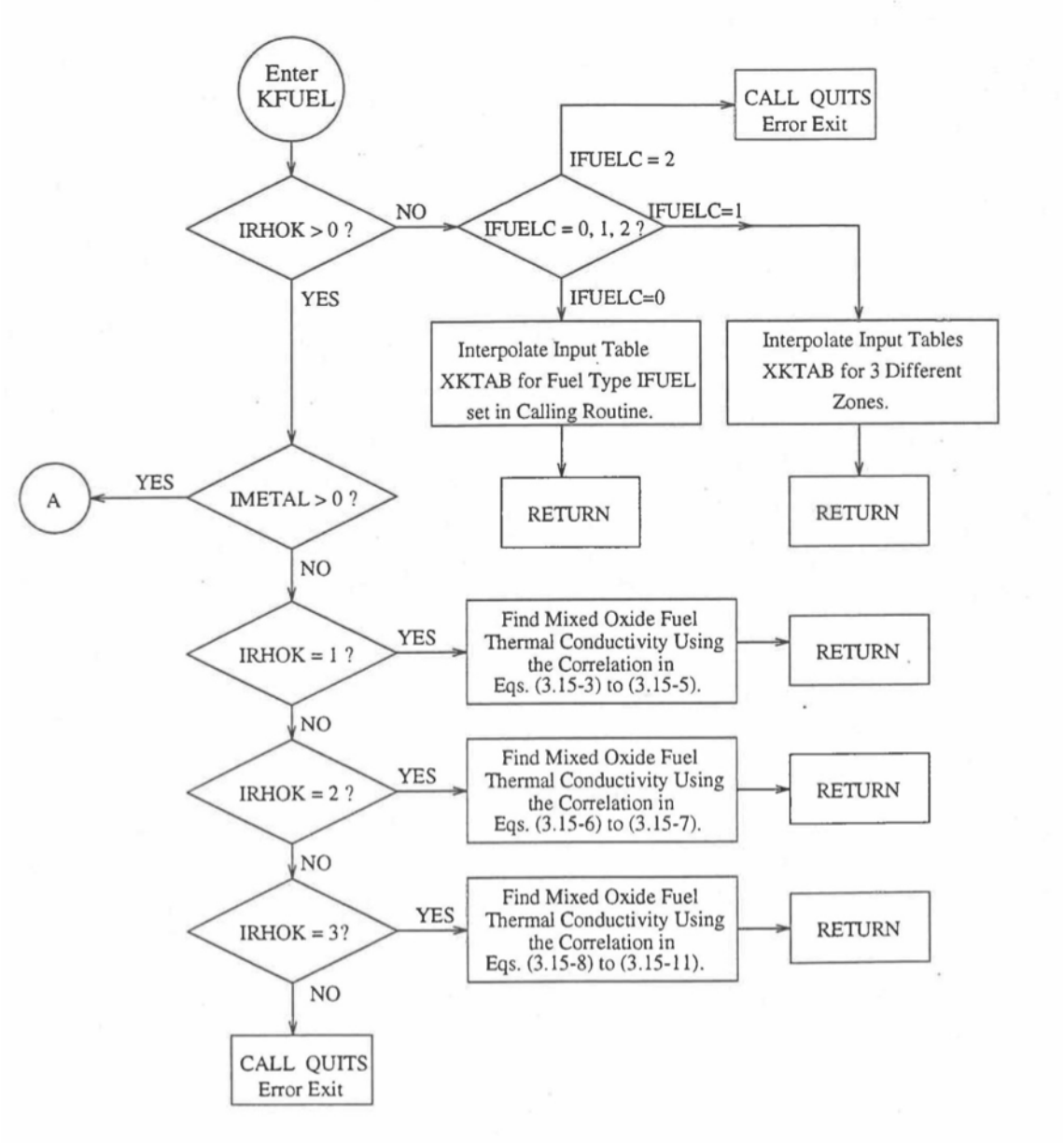


Figure 10.3.5: Flow Diagram of Subroutine KFUEL for Thermal Conductivity of All Fuel Radial Nodes in a Pin Axial Segment.

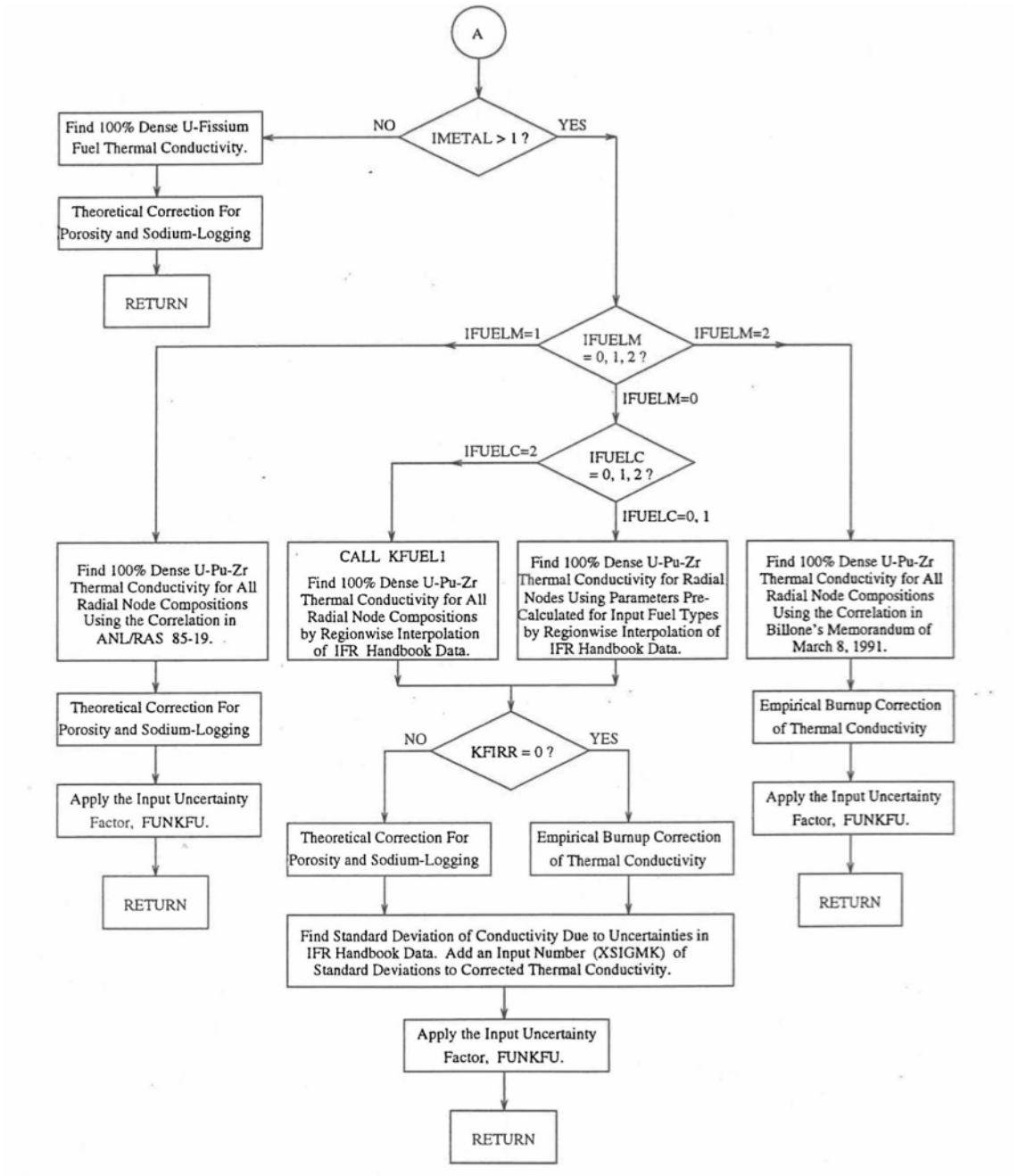


Figure 10.3.6: Flow Diagram of Subroutine KFUEL for Thermal Conductivity of All Fuel Radial Nodes in a Pin Axial Segment (Cont'd).

10.3.1 Temperature-Dependent Tables

The input parameter IRHOK=0 in a channel implies that temperature-dependent tables of specific heat, theoretical density and thermal conductivity of U-Pu-Zr alloy fuels of a maximum of 8 different alloy compositions (called fuel types in the input description given in Chapter 2) have been prepared by the user and supplied to the code as input data, and one of these tables is to be used in the channel. The core and blanket fuel types to be used in a channel are specified through the input variables IFUELV and IFUELB if the channel has a single radial fuel zone (indicated by the input parameter IFUELC=0). The fuel types are specified through the input array MFTZN if the channel has multiple radial fuel zones (indicated by the input parameter IFUELC=1). The compositions of the U-Pu-Zr metallic alloy fuel types are specified through the input array PUZRTP.

10.3.2 Regionwise Interpolation of IFR Handbook Data

SAS4A/SASSYS-1 is being developed to perform calculations for the Mark-V fuel (U-20Pu-10Zr) and other metallic alloy fuels of different compositions, especially the different fuel compositions across the pin radius resulting form the migration of fuel constituents during steady-state reactor operation. There is thus a need for modeling and correlation development for the thermal properties of U-Pu-Zr alloy fuels as a function of composition. This is so because the IFR Metallic Fuels Handbook [10-8, 10-9] presents thermal properties data for the 3 constituents and certain specific alloys only, and does not always present correlations with their coefficients and transition temperatures as functions of composition. Using the thermal properties data presented in the IFR Metallic Fuels Handbook, a method of regionwise interpolation in the U-Pu-Zr ternary composition triangle has been developed to estimate thermal properties of U-Pu-Zr alloy fuels as a function of composition and temperature. This method of evaluating fuel thermal properties is specified by setting the input parameter IRHOK=1 and the input parameter IFUELM=0 in a channel, and this is the recommended option among the built-in fuel thermal properties options.

Figure 10.3.7 shows the flow diagram of a subroutine PRECAL that is called by the input driver routine INPDRV (see flow diagram of Figure 2.2.2) to pre-calculate, using the method of regionwise interpolation, certain parameters (described below) stored for fast evaluation of U-Pu-Zr alloy fuel thermal properties by fuel type. Fuel thermal properties by fuel type can be used only in case of a single radial fuel zone (IFUELC=0), or multiple (i.e., up to 3) radial fuel zones (IFUELC=1), but not when each fuel radial mesh interval is a unique zone of a different alloy composition (IFUELC=2). In the latter case, fuel thermal properties are evaluated directly using the fuel compositions of the mesh intervals. The pre-calculated parameters by fuel type evaluated by the subroutine PRECAL are not used when IFUELC=2.

10.3.2.1 Enthalpy of U-Pu-Zr Fuel

10.3.2.1.1 Introduction

The IFR Metallic Fuels Handbook presents enthalpy and specific heat data for the 3 constituents and certain specific alloys only, and does not present correlations with their coefficients and transition temperatures as functions of composition.

Earlier, before the preparation of the Metallic Fuels Handbook [10-9], a method was developed to estimate the composition-dependent enthalpy of solid U-Pu-Zr alloy fuel by mass-weighted averaging of the enthalpy data of the 3 constituents, i.e., U, Pu and Zr metals [10-4]. Each of the constituent metals undergoes certain crystallographic phase transitions in the solid state that require considerable heat addition (causing an enthalpy jump) at the transition temperatures, but the alloy itself has no sharp transition temperature or enthalpy jump. A drawback of this method [10-4] of estimating U-Pu-Zr enthalpy is that the enthalpy jumps in the constituent metals data are required to be spread over an arbitrarily chosen temperature range containing the transition

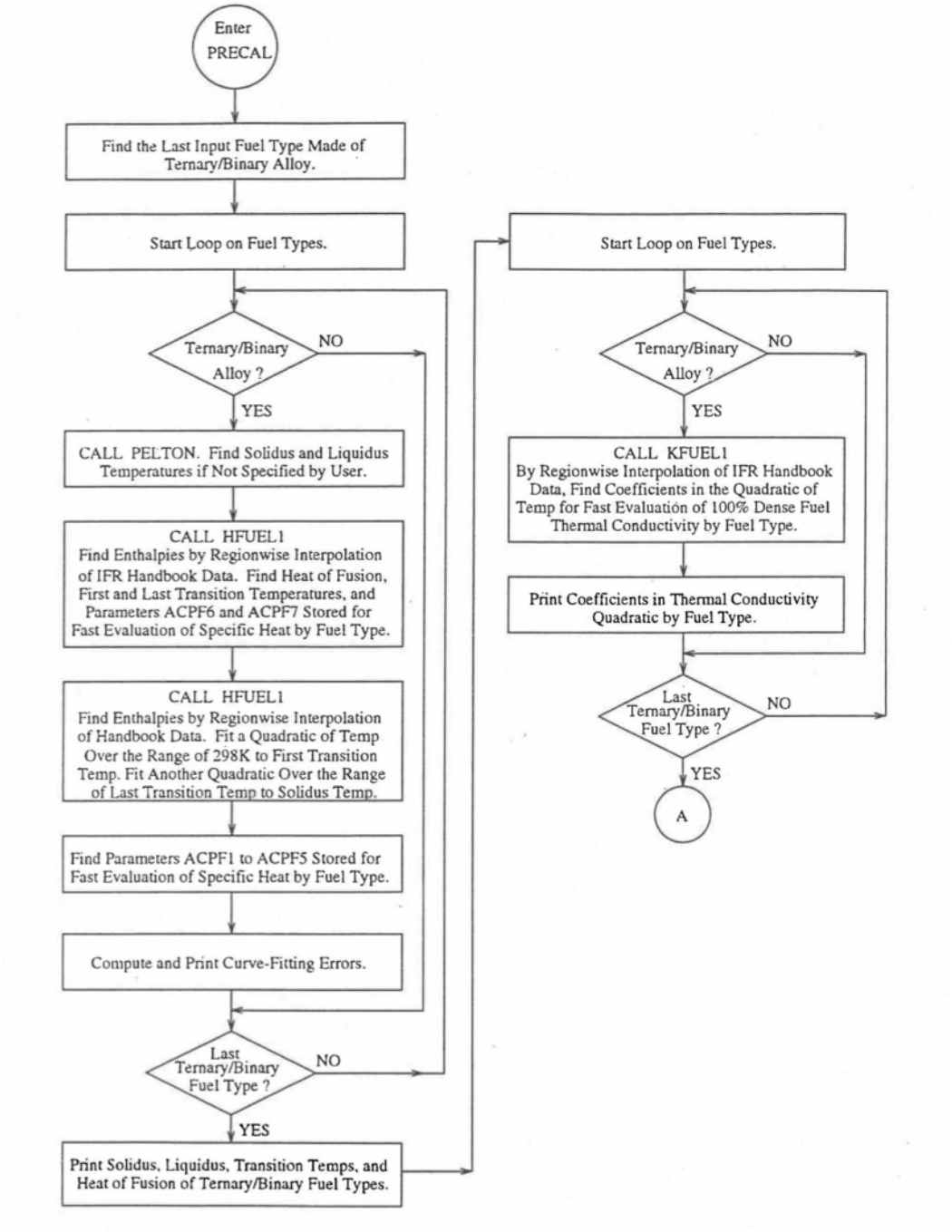


Figure 10.3.7: Flow Diagram of Subroutine PRECA1 to Pre-Calculate Parameters Stored for Fast Evaluation of U-Pu-Zr Specific Heat, Theoretical Density and Thermal Conductivity by Fuel Type.

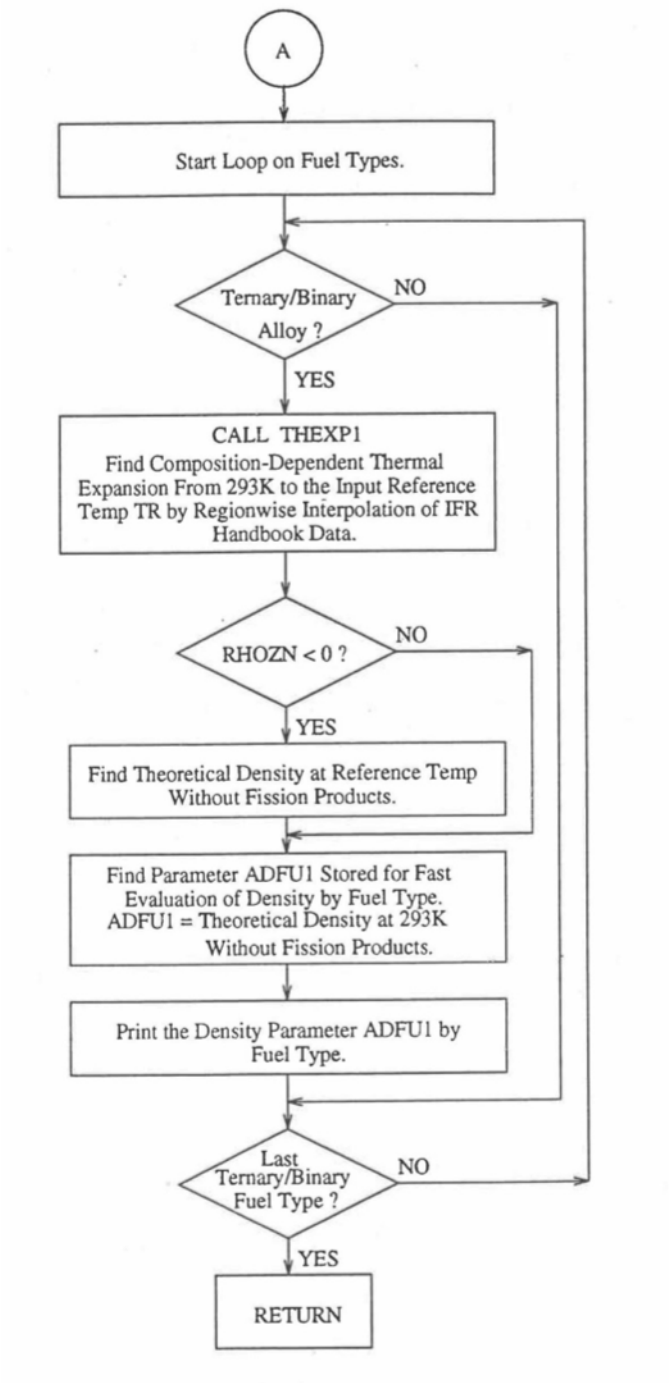


Figure 10.3.8: Flow Diagram of Subroutine PRECA1 to Pre-Calculate Parameters Stored for Fast Evaluation of U-Pu-Zr Specific Heat, Theoretical Density and Thermal Conductivity by Fuel Type (Cont'd).

temperatures. Another drawback is that the enthalpy data for the constituent metals was obtained from a source [10-10] other than the Metallic Fuels Handbook.

The method that is used in the FPIN2 code [10-11] to estimate the composition-dependent specific heat of U-Pu-Zr alloy fuel is also based on mass-weighted averaging of the 3 constituent specific heats. In this method, each constituent specific heat is assumed to be constant over the range of room temperature to the melting point of the constituent metal; and this assumption is tantamount to spreading the enthalpy jumps (due to solid-state phase transitions) over the same temperature range.

Recently, Billone has suggested that the enthalpy of solid U-Pu-Zr alloy fuel as a function of composition could be obtained by mixing the following 3 alloys rather than the 3 constituent metals [10-12]:

(1) U-15 wt % Pu-10 wt % Zr ternary alloy.

(2) U-(10 ± 5) wt % Zr binary alloy, and

(3) Pu metal.

The enthalpy data of U-15Pu-10Zr ternary, U-10Zr binary and Pu metal have been reported in the Metallic Fuels Handbook. The enthalpy of U-Zr binary alloy as a function of Zr content (in the neighborhood of 10 wt %) has also been correlated [10-12] by Billone using data for some binary alloys reported in the Metallic Fuels Handbook, i.e., 87 at. % U-Zr (U-5.4 wt. % Zr), U-10 wt % Zr, 59 at. % U-Zr (U-21.1 wt % Zr), and 39 at. % U-Zr (U-37.6 wt % Zr). The suggestion of mixing the 3 alloys rather than the 3 constituents provides a number of benefits. It removes the drawback of spreading the enthalpy jumps (due to solid-state phase transitions) over an arbitrarily chosen temperature range. The heats of solid-state transitions are naturally spread over the correct temperature range and included in the enthalpy data of the alloys that are mixed. The suggestion also improves the accuracy of the calculated enthalpy of the desired U-Pu-Zr alloy by choosing the 3 basic alloy compositions (that are mixed) in the neighborhood of the desired alloy composition.

Based on (a) this suggestion for obtaining the enthalpy of solid U-Pu-Zr alloy fuel, and (b) the assumption of ideal solution of molten U, Pu and Zr (used in the Metallic Fuels Handbook to obtain U-10Zr and U-15Pu-10Zr enthalpies at and above their liquidus temperatures) for obtaining the enthalpy of liquid U-Pu-Zr fuel, a method of regionwise interpolation and a subroutine HFUEL1 based on it have been developed and incorporated into the SAS4A/SASSYS-1 codes. This method exactly reproduces enthalpy data in the Handbook for molten U-15Pu-10Zr, U-10Zr, U, Pu and Zr liquids. The enthalpy data in the Handbook for solid U-15Pu-10 Zr, U-5.4Zr, U-10Zr, U-21.1Zr, U-37.6Zr alloys, and solid U and Pu metals are also accurately reproduced by this method.

10.3.2.1.2 Description of the Method

Figure 10.3.9 shows the U-Pu-Zr ternary system composition triangle. To perform design and safety calculations for the Mark-V fuel and other metallic alloy fuels important in the IFR program, including the different fuel compositions across the pin radius caused by migration of fuel constituents during reactor operation, it is sufficient to be able to evaluate enthalpy of alloy compositions (represented by points) located in a small region, equal in area to about one-third of the whole composition triangle, in the U corner of the triangle. The present method models and evaluates the enthalpy of all alloy compositions located in the triangular region UEPu of Figure 10.3.9.

The enthalpy of a molten alloy (at and above its liquidus temperature) as a function of composition is evaluated in the present method by assuming the molten alloy to be an ideal solution of the 3 liquid components U, Pu and Zr. The enthalpy evaluation approach for the molten alloy is changed from that for the solid alloy because (1) the IFR Metallic Fuels Handbook uses this 3-component ideal solution assumption for evaluating enthalpy of molten U-10Zr, and U-15Pu-10Zr, and recommends this assumption for other alloy compositions, and (2) the solid-state phase structure is lost after melting, i.e., the liquid obtained after melting is the same irrespective of the solid phase which melted; the alternate approach of arriving at the enthalpy at liquidus by adding the heat of fusion and the sensible heat (required for raising temperature form solidus to liquidus) to the enthalpy

at solidus contains uncertainties which can be avoided by using the 3-liquid ideal solution assumption. Between the solidus and liquidus temperatures, the enthalpy of the alloy is linearly interpolated as recommended in the Metallic Fuels Handbook. The solidus and liquidus temperatures required in the present method may be either obtained from experimental measurements for the desired alloy, if available, or calculated using a subroutine developed by Pelton [10-13] and incorporated into the SAS4A/SASSYS-1 codes.

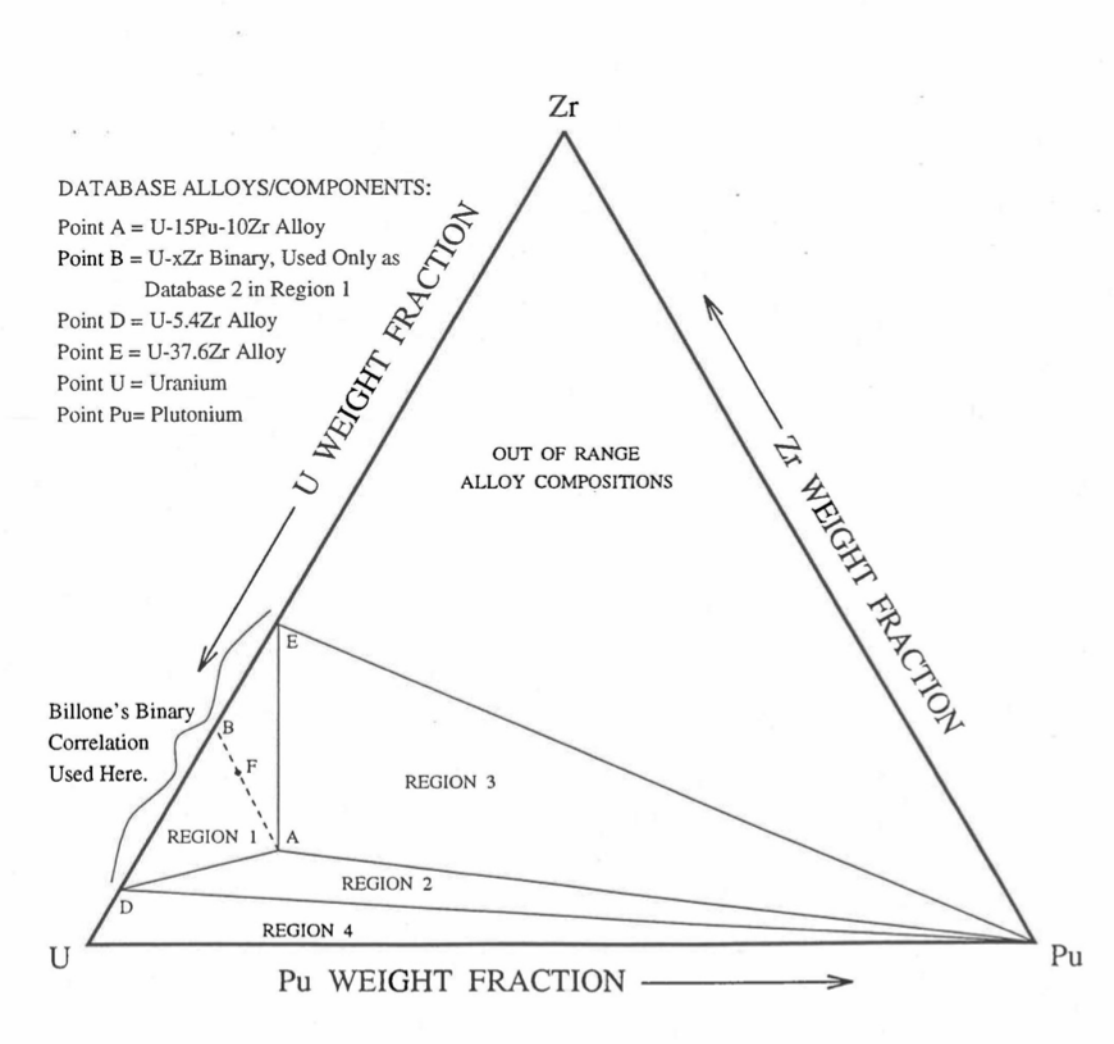


Figure 10.3.9: Diagram Showing Enthalpy Interpolation Regions for Solid U-Pu-Zr Ternary Alloy.

Base on the enthalpy data available in the IFR Handbook, the present method divides the triangular region UEPu, i.e., the region representing all those alloy compositions whose enthalpy in the solid state can be evaluated by the present method, into four regions as shown in Figure 10.3.9. There are data in the Handbook for solid U, Pu, U-15Pu-10Zr (represented by point A in Figure 10.3.9), and six U-Zr binary alloys with Zr content ranging form 5.4 wt % (point D) to 75.7 wt %. Because the IFR program deals with fuels having a limited amount of Zr, the highest Zr content U-Zr binary alloy data used in the method is that of U-37.6 wt % Zr (point E). In region 1 given by triangle ADE of Figure 10.3.9, the enthalpy of a solid U-Pu-Zr alloy of a desired composition represented by point F is obtained by mixing or linearly interpolating between the database alloys A (i.e., U-15Pu-10Zr) and B (i.e., the U-Zr binary defined by the point of intersection of line AF). The Zr content of binary alloy B is given in Section D, and the enthalpy of this binary alloy is found form a correlation by Billone [10-12] of all the enthalpy data in the Handbook for U-Zr alloys between points D and E. The coefficients of this correlation

are composition-dependent. By comparison with data in the Handbook, this U-Zr enthalpy correlation has been found to be accurate.

The enthalpy of a solid U-Pu-Zr alloy of a desired composition represented by a point inside region 2 (given by triangle ADPu of Figure 10.3.9 is obtained by mixing the 3 database alloys A, D (i.e., U-5.4Zr) and Pu metal. The enthalpy of a solid U-Pu-Zr alloy of a desired composition represented by a point inside region 3 (given by triangle AEPu) is obtained by mixing the 3 database alloys A, E (i.e., U-37.6Zr) and Pu as shown in Table 10.3.1. Regions 1, 2 and 3 are all in the vicinity of the important ternary fuel U-15Pu-10Zr, and in the present method the enthalpy data for this fuel has a proportional influence on the interpolated enthalpy for any desired alloy composition in these three regions. The enthalpy of a solid U-Pu-Zr alloy of a desired composition represented by a point inside region 4 (given by triangle UDPu) is obtained by mixing the database alloy D, and U and Pu metals. The enthalpy of U metal used in the present method is obtained from data and correlation given in the Handbook because an extrapolation of the U-Zr binary alloy enthalpy correlation does not produce an accurate U metal enthalpy. The enthalpies of the database alloys used in the interpolation are given in Section C. The proportions or weight fractions of the 3 database alloys in mixing are given in Section D.

Between the first and the last solid-state phase transition temperatures, 873 K and 923 K, the specific heat of database alloy A (i.e., U-15Pu-10Zr) is much higher than the specific heat above and below this temperature range. These transition temperatures, denoted by T_α and T_γ, are required for the other database alloys and the desired U-Pu-Zr alloy in order to accurately account for the specific heat variation over this temperature range. In the present method, these transition temperatures for the U-Zr binary alloys used as database, and for all U-Pu-Zr ternary alloy compositions in regions 1, 2 and 3 are assumed to be equal to those of database alloy A because of lack of composition-dependent correlations for the transition temperatures. Such an assumption was suggested by Billone [10-12] for U-Pu-Zr alloys in the vicinity of U-15Pu-10Zr, and for U-Zr binary alloys used in mixing calculations. The choice of T_α and T_γ for Pu metal is immaterial because its enthalpy data, after the simplification suggested by Billone [10-12] and described in Section C, does not have any temperature range of abruptly high specific heat. In region 4 of Figure 10.3.9, the average specific heat of U metal between its first and last transition temperatures is abruptly high, and therefore, T_α is interpolated between the value of T_α (942 K) for U metal [10-8] and the assumed value for other databases (873 K); and T_γ is interpolated between the value of T_γ (1049 K) for U metal [10-8] and the assumed value for other databases (923 K). These transition temperatures are summarized in Table 10.3.1.

Section C describes the enthalpy correlations for the basic alloys and their comparison with the data in the Handbook. Section D describes the equations used for interpolating between the database alloys. Section E summarizes the variation of fuel enthalpy with composition computed using the subroutine HFUEL1 based on this method.

Table 10.3.1: Basic Alloys Whose IFR Handbook Enthalphy Data are Combined to Obtain the Enthalpy of a Desired U-Pu-Zr Alloy Composition

	Database Alloys				
Region[a]	1	2	3	Desired Alloy T_α, K^b	Desired Alloy T_γ, K^b
Region 1	U-15Pu-10Zr	U-xZr[c]	not required	873	923
Region 2	U-15Pu-10Zr	U-5.4455Zr	Pu metal	873	923
Region 3	U-15Pu-10Zr	U-37.6105Zr	Pu metal	873	923
Region 4	U metal	U-5.4455Zr	Pu metal	873 to 942[d]	923 to 1049[e]

[a] These regions are shown on the U-Pu-Zr ternary composition triangle in Figure 10.3.9.

[b] T_α and T_γ denote the first and the last solid-state phase transition temperatures of the alloy. T_α and T_γ of alloy compositions in the vicinity of U-15Pu-10Zr are assumed to be equal to those of U-15Pu-10Zr, i.e., 873 K and 923 K, as suggested by Billone [10-12].

[c] The weight fraction x of Zr is selected such that the point representing U-xZr binary lies on the line joining the points representing the desired alloy composition and U-15Pu-10Zr database alloy on the ternary composition triangle shown in Figure 10.3.9.

[d] Interpolated between 942 K (the value for U metal) and 873 K (the assumed value for other databases, see footnote b).

[e] Interpolated between 1049 K (the value for U metal) and 923 K (the assumed value for other databases, see footnote b).

10.3.2.1.3 Enthalpy Correlations of Database Alloys

Figure 10.3.10 to Figure 10.3.16 show the temperature dependence of the enthalpy of all the basic alloys used in the present method. The difference in behavior of a database alloy when mixed to produce a desired alloy composition, e.g., U-20Pu-20Zr, as compared to its behavior in isolation is also clarified by these figures. For this purpose, the comparison of enthalpy of the basic alloy in isolation, evaluated from the correlations used in the present method, with the data in the Metallic Fuels Handbook [10-8] is also shown. These plots cover the entire temperature range from 298 K (zero enthalpy reference temperature) to 2000 K, a temperature well beyond the liquidus. In the following sections, the enthalpy correlations of the database alloys for use in solid and liquid phases are separately described.

1. *Database Used in Liquid Phase*

The enthalpy of a molten U-Pu-Zr alloy fuel (at and above its liquidus temperature) is evaluated by assuming the molten alloy to be an ideal solution of the 3 liquid components U, Pu and Zr. Thus the enthalpies of the 3 liquid components as functions of temperature from the database. Based on the data for enthalpy at U liquidus and specific heat of liquid U given in the Handbook [10-8], the enthalpy of liquid U can be written as

$$H_{\text{lu}} = 58347 + 48.66\,(T - 1408) \tag{10.3-1}$$

where

H_{lu} = enthalpy of liquid U, J/mol

T = temperature of liquid U, K

58347 = liquid U enthalpy (J/mol) at its melting point 1408 K,

48.66 = liquid U specific heat, J/mol-K.

Eq. (10.3-1) simplifies to

$$H_{\text{lu}} = 48.66T - 10166.28 \tag{10.3-2}$$

Similarly, based on the data for enthalpy at liquidus and specific heat of the liquid phase given in the Handbook for Pu and Zr, the following equations for liquid phase enthalpy are obtained:

$$H_{\text{lp}} = 42.258T - 7291.55 \tag{10.3-3}$$

$$H_{\text{lz}} = 33.5T + 6116.5 \tag{10.3-4}$$

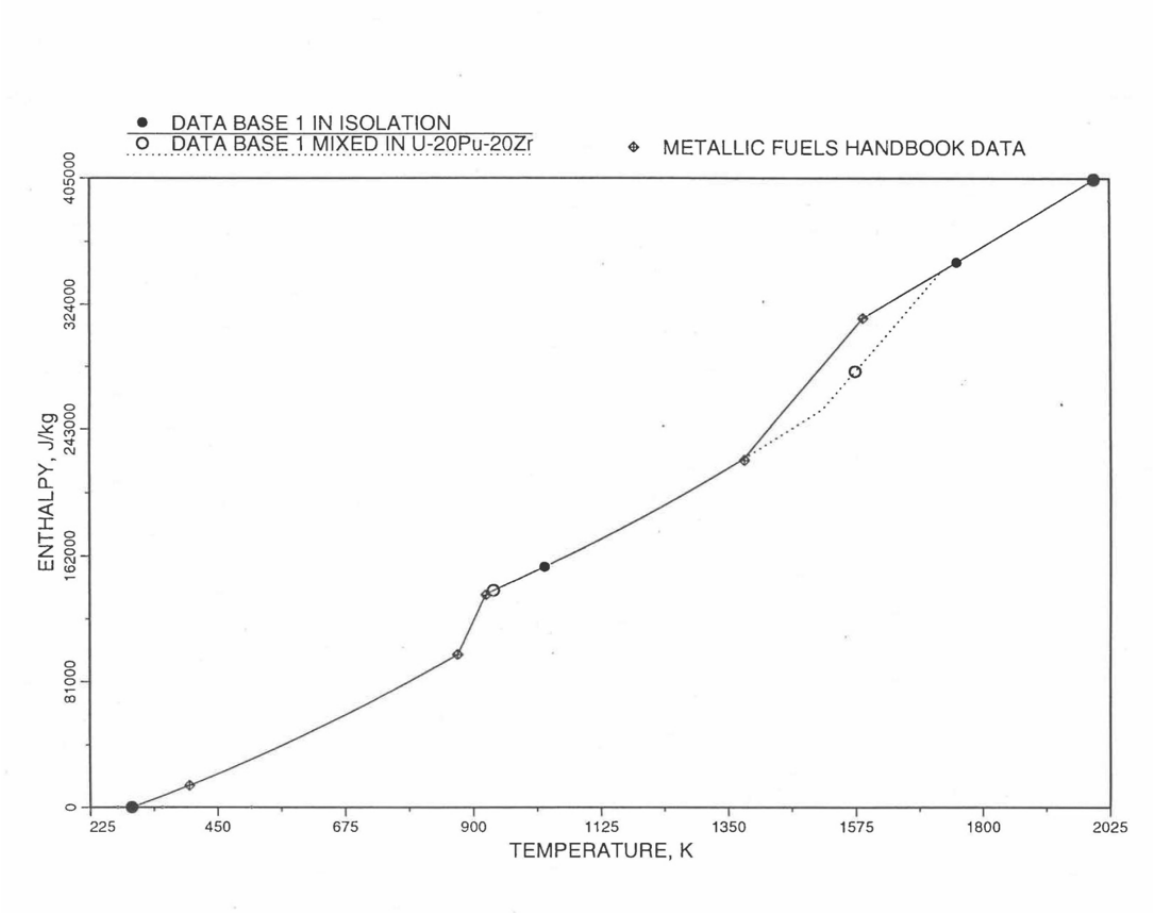


Figure 10.3.10: Enthalpy of U-15Pu-10Zr Obtained from Eq. (10.3-2) to Eq. (10.3-5) That is Used as Database 1 in Regions 1, 2 and 3; and its Comparison with Metallic Fuels Handbook Data.

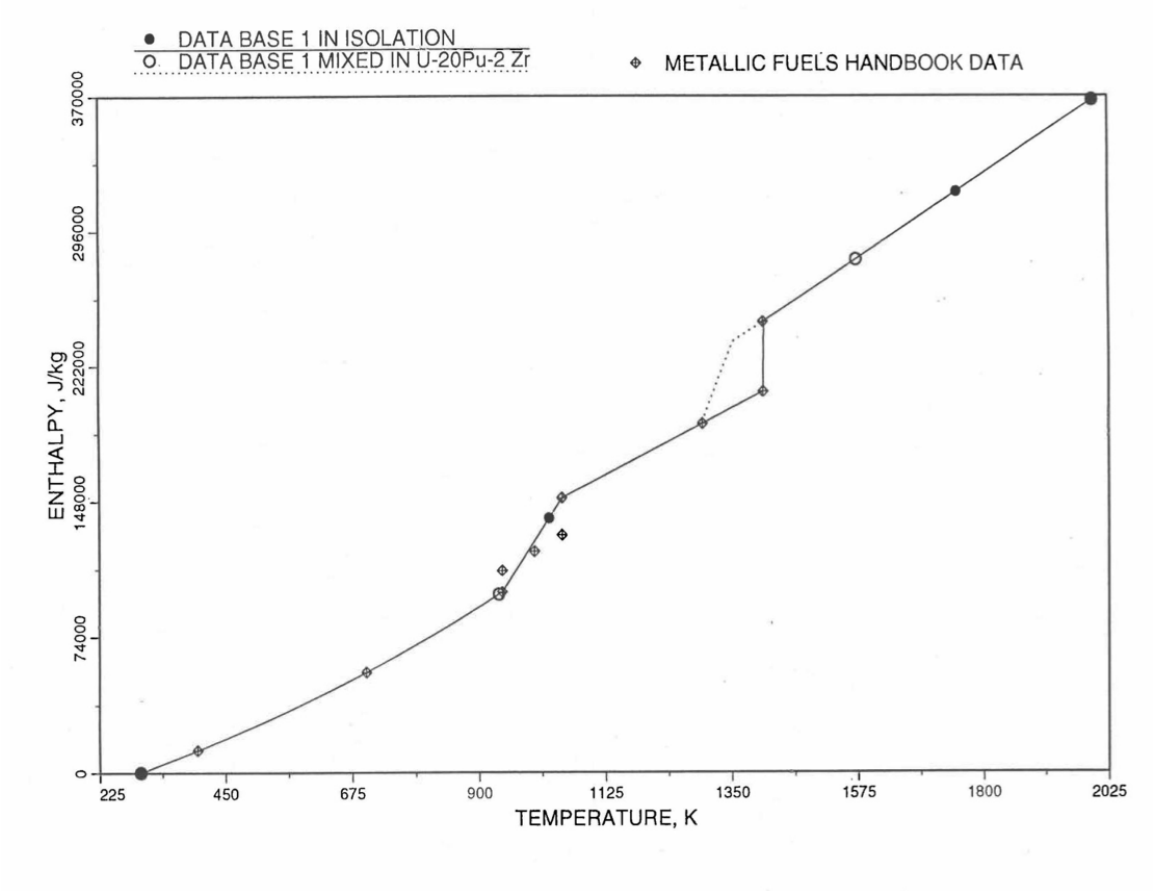


Figure 10.3.11: Enthalpy of U Metal Obtained From Eq. (10.3-2) and Eq. (10.3-6) That is used as Database 1 in Region 4; and its Comparison With Metallic Fuels handbook Data.

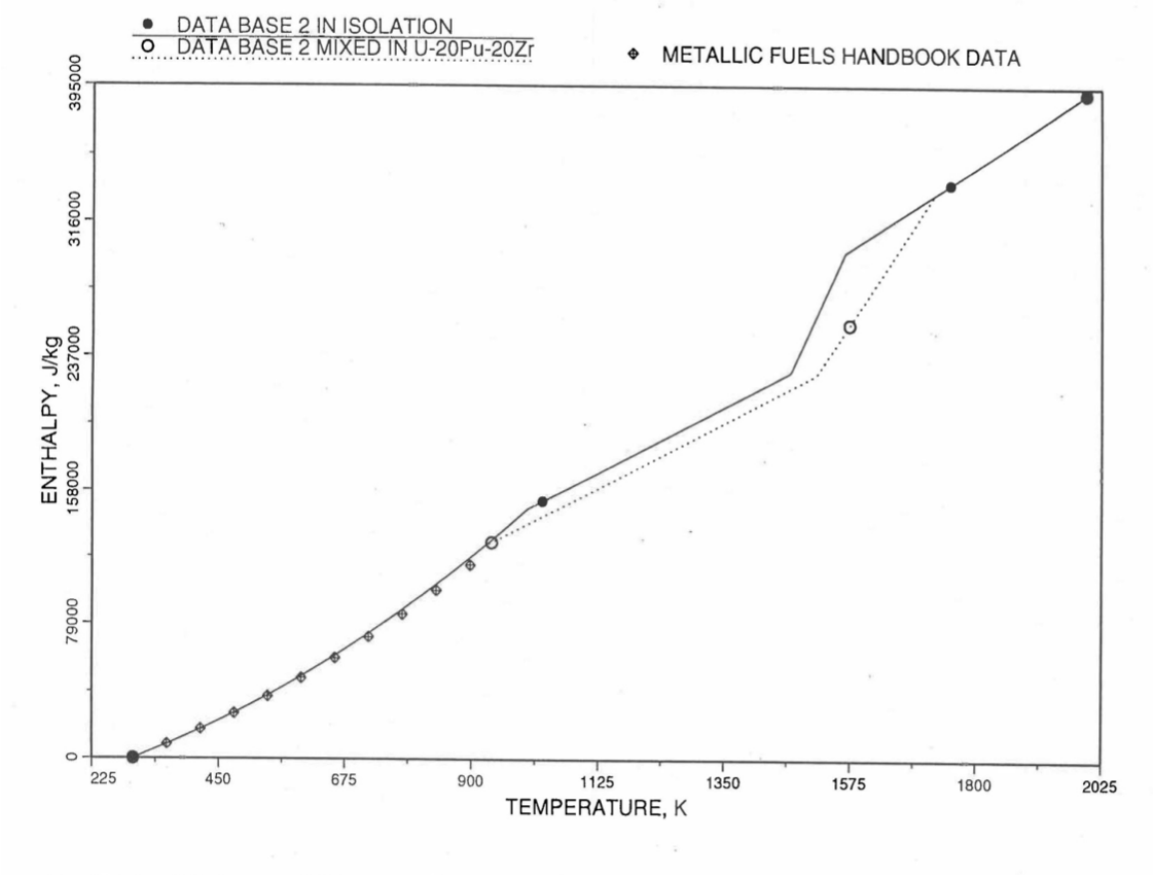


Figure 10.3.12: Enthalpy of U-5.4Zr Obtained From Eq. (10.3-2), Eq. (10.3-4) and Eq. (10.3-7) That is Used as Database 2 in Regions 1, 2 and 4; and its Comparison With Metallic Fuels Handbook Data.

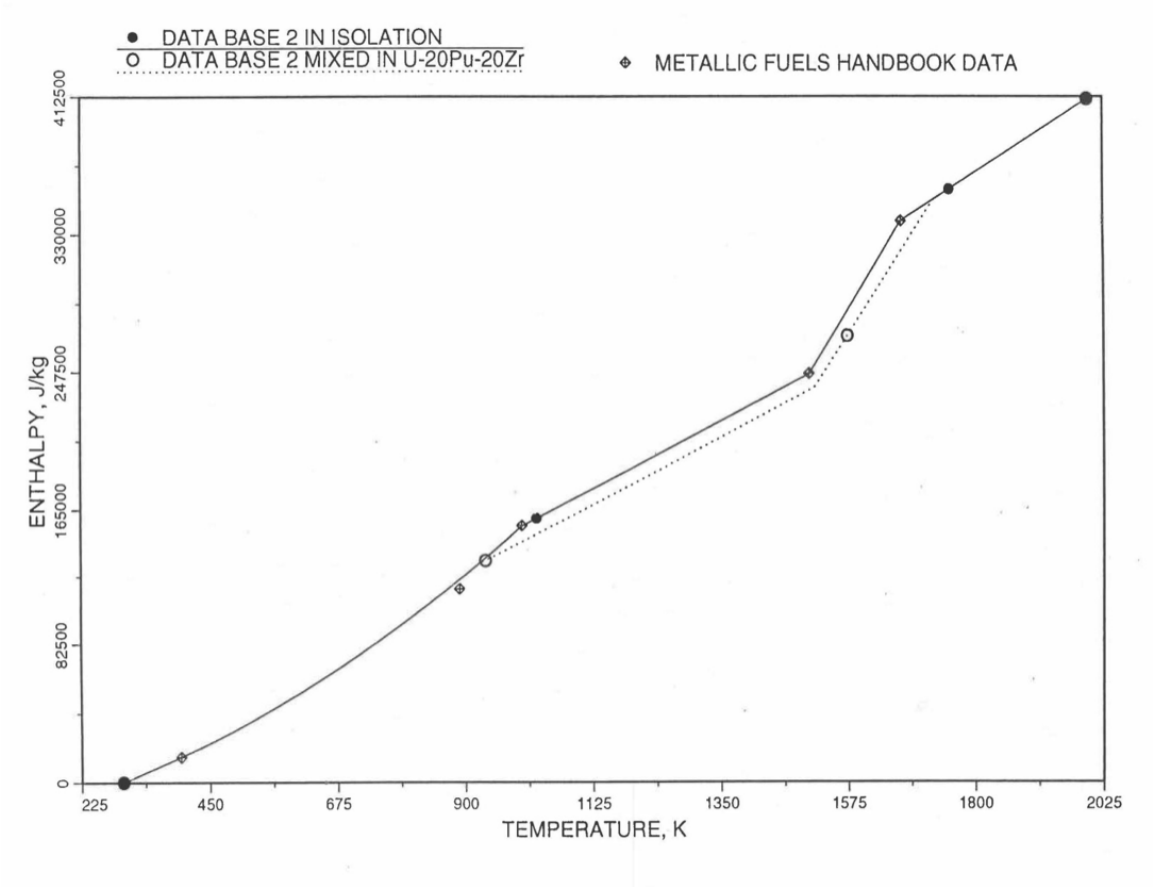


Figure 10.3.13: Enthalpy of U-10Zr Obtained from Eq. (10.3-2), Eq. (10.3-4) and Eq. (10.3-7) That is Used as Database 2 in Regions 1, 2 and 4; and its Comparison With Metallic Fuels Handbook Data.

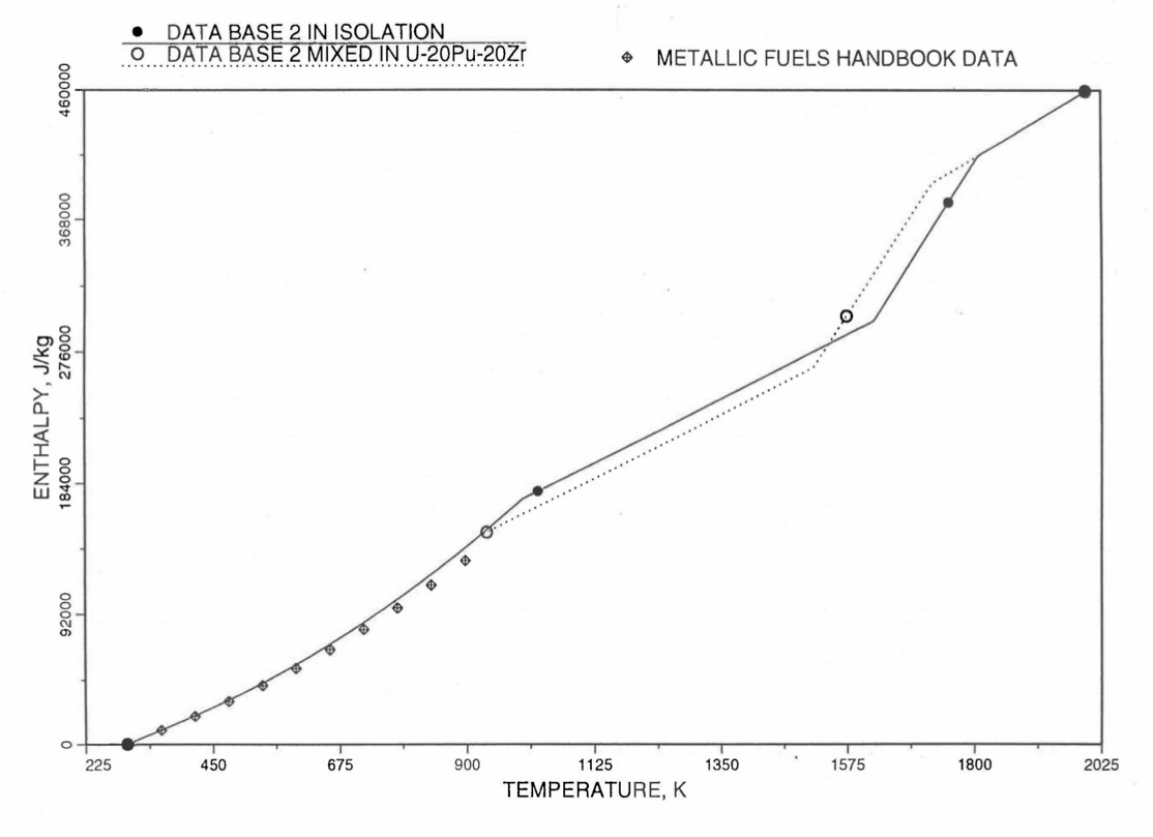


Figure 10.3.14: Enthalpy of U-21.1Zr Obtained From Eq. (10.3-2), Eq. (10.3-4) and Eq. (10.3-7) That is Used as Database 2 in Region 1; and its Comparison With Metallic Fuels Handbook Data.

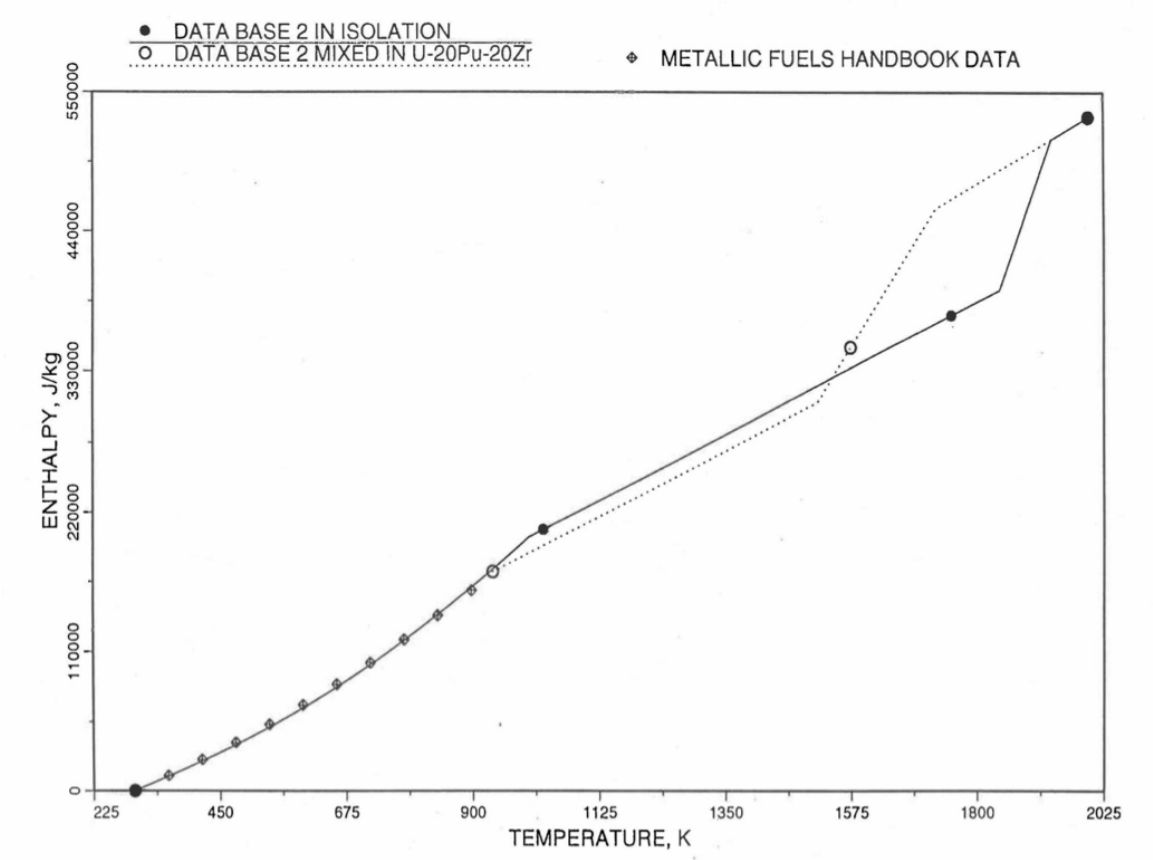


Figure 10.3.15: Enthalpy of U-37.6Zr Obtained From Eq. (10.3-2), Eq. (10.3-4) and Eq. (10.3-7) That is Used as Database 2 in Regions 1 and 3; and its Comparison With Metallic Fuels Handbook Data.

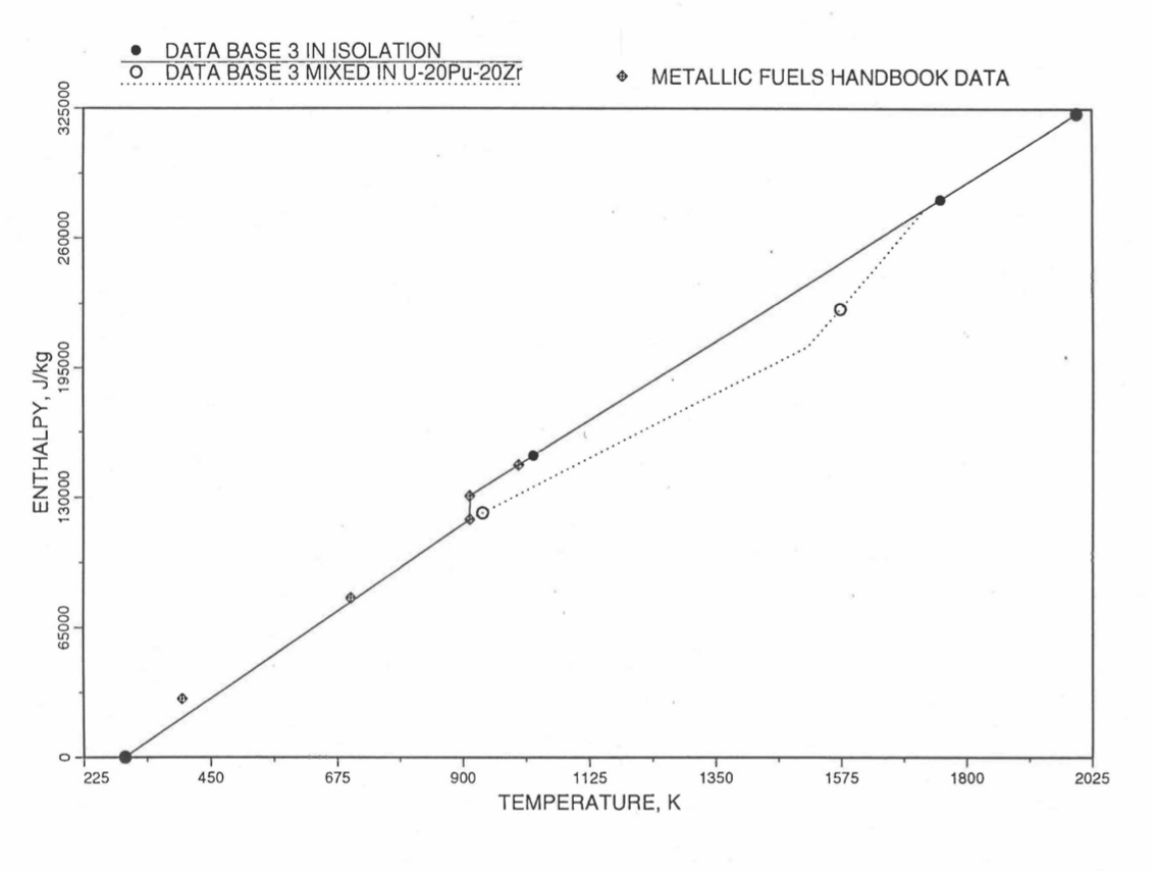


Figure 10.3.16: Enthalpy of Pu Metal Obtained From Eq. (10.3-3) and Eq. (10.3-8) That is Used as Database 3 in Regions 2, 3 and 4; and its Comparison With Metallic Fuels Handbook Data.

where

H_{lp}, H_{lz} = enthalpies of liquid *Pu* and liquid *Zr*, J/mol

T = temperature of the liquid phase, K.

When mixed to form a U-Pu-Zr molten mixture, Eq. (10.3-2) to Eq. (10.3-4) are used for all temperatures at and above the liquidus temperature of the U-Pu-Zr alloy, even if the liquidus temperature is below the melting point of U, Pu or Zr.

2. *Database 1 Used in Solid Phase*

As described in Section B, the ternary fuel U-15Pu-10Zr is used as database 1 in regions 1, 2 and 3 of Figure 10.3.9; and the U metal is used as database 1 in region 4. The enthalpy data and correlation for U-15Pu-10Zr presented in the Metallic Fuels Handbook [10-8] and in the memorandum by Billone [10-12] can be written as

$$H_{\text{a}}(T) = \begin{cases} 19.34\,(T-298) + 0.0133\left(T^2 - 298^2\right) & T \le 873\text{ K} \quad \text{(10.3-5a)} \\ H_{\text{a}}(873) + 162\,(T-873) & 873 < T \le 923\text{ K} \quad \text{(10.3-5b)} \\ H_{\text{a}}(923) + 8.752\,(T-923) + 0.01304\left(T^2 - 923^2\right) & 923 < T \le 1379\text{ K} \quad \text{(10.3-5c)} \\ H_{\text{a}}(1379) + 89.95\,(T-1379) & 1379 < T \le 1588\text{ K} \quad \text{(10.3-5d)} \\ H_{\text{a}}(1588) + 44.4\,(T-1588) & T > 1588\text{ K} \quad \text{(10.3-5e)} \end{cases} \tag{10.3-5}$$

where

H_{a} = enthalpy of U-15Pu-10Zr alloy, J/mol

T = temperature of the alloy, K

Here 1379 K and 1588 K are the solidus and liquidus temperatures of this alloy. In the present method, Eq. (10.3-5) and Eq. (10.3-5) are never required. Eq. (10.3-5) covers the temperature range containing solid-state phase transitions. When mixed to form a desired U-Pu-Zr alloy composition, Eq. (10.3-5) for the enthalpy in the temperature range above the solid-state phase transitions is extended up to the solidus temperature of the desired alloy. This difference in behavior of U-15Pu-10Zr when mixed to produce a desired alloy composition, e.g., U-20Pu-20Zr, as compared to its behavior in isolation is shown in Figure 10.3.10.

The enthalpy data and correlation for U metal (used as database 1 in region 4) presented in the Metallic Fuels Handbook [10-8] can be written as

$$H_{\text{u}}(T) = \begin{cases} 26.92\,(T-298) - 1.251\times10^{-3}\left(T^2 - 298^2\right) + \\ 8.852\times10^{-6}\left(T^3 - 298^3\right) - 7.7\times10^{4}\left[\dfrac{1}{298} - \dfrac{1}{T}\right] & T \le 942\text{ K} \quad \text{(10.3-6a)} \\ H_{\text{u}}(942) + 113.467\,(T-942)\,, & 942 < T \le 1049\text{ K} \quad \text{(10.3-6b)} \\ H_{\text{u}}(1049) + 38.284\,(T-1049)\,, & 1049 < T \le 1408\text{ K} \quad \text{(10.3-6c)} \\ H_{\text{u}}(1408) + 9142 + 48.66\,(T-1408)\,, & T > 1408\text{ K} \quad \text{(10.3-6d)} \end{cases} \tag{10.3-6}$$

where

$H_{\text{u}}(T)$ = enthalpy (J/mol) of U metal at temperature T(K).

Here 942 K and 1049 K are the first and the last solid-state phase transition temperatures of U metal, and 1408 K is the melting point. The average specific heat [10-8] over the solid-state phase transition range is 113.467 J/mol-K as used in Eq. (10.3-6). In the present method, Eq. (10.3-6) is never required. When mixed to form a desired U-Pu-Zr alloy composition, Eq. (10.3-6) for the enthalpy in the temperature range above the solid-state phase transitions is extended up to the solidus temperature of the desired alloy. This difference in behavior of U metal when mixed to produce a desired alloy composition, e.g., U-20Pu-2Zr, as compared to its behavior in isolation is shown in Figure 10.3.11.

3. *Database 2 Used in Solid Phase*

As described in Section B, U-Zr binary alloys of 5.4 to 37.6 wt % Zr content are used as database 2 in different regions as given in Table 10.3.1. The enthalpy data and correlations for 87 at.% U-Zr (U-5.4wt %Zr), U-10 wt %Zr, 59 at.% U-Zr (U-21.1wt %Zr), and 39 at.% U-Zr (U-37.6wt %Zr) presented in the Metallic Fuels Handbook have been correlated by Billone [10-12] into the following single equation with composition-dependent coefficients. Here the relationships between atom fractions and weight fractions were derived using an average atomic weight of 236.678 for U (based on a U-235 enrichment of 0.478 weight fraction as done by Billone for Mark-V fuel), an atomic weight of 91.22 from Zr, and an atomic weight of 239.13 for Pu.

$$H_{\text{uz}}(T) = \begin{cases} 1.594\,(A_{\text{u}} + 0.3219A_{\text{z}})\,(T - 298) \\ \quad + 0.03235\,(A_{\text{u}} + 0.8495A_{\text{z}})\,\left(T^2 - 298^2\right) \\ \quad + 928700\,(A_{\text{u}} + 1.0243A_{\text{z}})\left(\dfrac{1}{298} - \dfrac{1}{T}\right), & T \le T_\gamma \quad \text{(10.3-7a)} \\ H_{\text{uz}}(T_\gamma) + 38.28\,(A_{\text{u}} + 0.8203A_{\text{z}})\,(T - T_\gamma), & T_\gamma < T \le T_{\text{s}} \quad \text{(10.3-7b)} \\ H_{\text{uz}}(T_{\text{s}}) + 9142\,(A_{\text{u}} + 5.938A_{\text{z}})\,(T - T_{\text{s}})\,/\,(T_{\text{l}} - T_{\text{s}}), & T_{\text{s}} < T \le T_{\text{l}} \quad \text{(10.3-7c)} \\ H_{\text{uz}}(T_{\text{l}}) + 48.66\,(A_{\text{u}} + 0.6885A_{\text{z}})\,(T - T_{\text{l}}), & T > T_{\text{l}} \quad \text{(10.3-7d)} \end{cases} \tag{10.3-7}$$

where

$H_{\text{uz}}(T)$ = enthalpy (J/mol) of the U-Zr binary alloy at temperature T (K),

$A_{\text{u}}, A_{\text{z}}$ = atom fractions of U and Zr in the alloy,

T_γ = the highest solid-state phase (γ-phase) transition temperature,

$T_{\text{s}}, T_{\text{l}}$ = solidus and liquidus temperatures of the U-Zr binary alloy.

For the U-Zr binary alloy, the temperatures T_γ, T_{s} and T_{l} should be taken form the U-Zr phase diagram. For U-10 wt% Zr, the temperatures T_γ, T_{s} and T_{l} are 1000 K, 1506 K and 1669 K. In the present method, Eq. (10.3-7) and Eq. (10.3-7) are never required. When mixed to form a desired U-Pu-Zr alloy composition, the temperatures T_γ, T_{s} and T_{l} for the U-Pu-Zr ternary alloy are to be used in Eq. (10.3-7) as suggested by Billone [10-12]. In the present method, this amounts to (1) lowering T_γ from the U-Zr value of about 1000 K to the U-Pu-Zr alloy value of 923 K (see Table 10.3.1), and (2) extending Eq. (10.3-7) for U-Zr enthalpy in the γ-phase temperature range to the solidus temperature of the desired U-Pu-Zr alloy composition. This difference in behavior of four U-Zr alloys (database 2 alloys) when mixed to produce a desired alloy composition, e.g., U-20Pu-20Zr, as compared to their behavior in isolation is shown in Figure 10.3.12 to Figure 10.3.15. These plots also show the agreement between the enthalpy of U-Zr alloys in isolation, evaluated from this correlation, with the data in the Metallic Fuels Handbook [10-8].

4. *Database 3 Used in Solid Phase*

As described in Section B, the Pu metal is used as database 3 in regions 2, 3, and 4. Region 1 does not require this database. The enthalpy data for Pu metal presented in the Metallic Fuels Handbook [10-8] has a number of solid-state phase transitions causing relatively small enthalpy jumps at the transition temperatures. As suggested by Billone [10-12], the following equation is a simplified representation of Pu metal enthalpy data [10-8], in the spirit of mixing to produce a desired U-Pu-Zr alloy composition (the value of the average Pu specific heat from the reference temperature of 298 K to Pu melting point of 913 K has been corrected):

$$H_{\text{p}}(T) = \begin{cases} 46.286\,(T - 298), & T \le 913K \quad \text{(10.3-8a)} \\ H_{\text{p}}(913) + 34.434\,(T - 913), & 913 < T \le T_{\text{s}} \quad \text{(10.3-8b)} \\ H_{\text{p}}(913) + 34.434\,(T - 913) + 2824\,(T - T_{\text{s}})\,/\,(T_{\text{l}} - T_{\text{s}}), & T_{\text{s}} < T \le T_{\text{l}} \quad \text{(10.3-8c)} \\ H_{\text{p}}(T_{\text{l}}) + 42.258\,(T - T_{\text{l}}) & T > T_{\text{l}} \quad \text{(10.3-8d)} \end{cases} \tag{10.3-8}$$

where

$H_{\text{p}}(T)$ = enthalpy (J/mol) of Pu metal at temperature T,

913 K = the melting point of Pu metal

T_s, T_l = solidus and liquidus temperatures of the desired U-Pu-Zr alloy composition, and not of pure Pu metal.

Here 46.286 J/mol-K is the average Pu specific heat between 298 K and 913 K, and 34.434 J/mol-K is the specific heat of the last solid-stat phase (ε-phase) of Pu metal that is extended up to the liquidus temperature of the desired U-Pu-Zr alloy composition to account for the sensible heat absorbed by Pu. Eq. (10.3-8) indicates that the Pu heat of fusion required for solid-to-liquid phase transition is added linearly between the solidus and liquidus temperatures of the desired U-Pu-Zr alloy composition. In the present method, Eq. (10.3-8) and Eq. (10.3-8) are never required. The difference in behavior of Pu metal when mixed to produce a desired alloy composition, e.g., U-20Pu-20Zr, as compared to its behavior in isolation is shown in Figure 10.3.16. This plot also shows the comparison between the enthalpy of Pu metal in isolation with the date in Metallic Fuels Handbook [10-8]. In Figure 10.3.16, the Pu enthalpy above the solidus temperature of the desired U-Pu-Zr alloy composition was obtained by linearly joining the enthalpy at solidus evaluated from Eq. (10.3-8) with the enthalpy at liquidus evaluated from Eq. (10.3-3) as mentioned in Section B.

It should be noted that the contribution of Pu enthalpy to U-Pu-Zr fuel compositions in the vicinity of U-15Pu-10Zr (point A of Figure 10.3.9) is small because the corner Pu of the ternary composition triangle is far removed from point A and the point representing the desired fuel composition. The major contribution comes from U-15Pu-10Zr enthalpy.

10.3.2.1.4 Method of Interpolation Between Database

As described in Section B, the enthalpy of a desired U-Pu-Zr fuel composition at and above its liquidus temperature is evaluated by assuming it to be an ideal solution of the 3 liquid components U, Pu and Zr whose enthalpies are given by Eq. (10.3-2) to Eq. (10.3-4). This method is tantamount to mass-weighted averaging of the component enthalpies.

$$h_l(T) = 1000\left(W_u\frac{H_{lu}}{a_u} + W_p\frac{H_{lp}}{a_p} + W_z\frac{H_{lz}}{a_z}\right) \tag{10.3-9}$$

where

$h_l(T)$ = enthalpy (J/kg) of a U-Pu-Zr fuel composition at a temperature T, at and above its liquidus,

W_u, W_p, W_z = weight fractions of U, Pu and Zr in the U-Pu-Zr fuel composition

H_{lu}, H_{lp}, H_{lz} = enthalpies (J/mol) of U, Pu and Zr metals in the liquid phase

a_u, a_p, a_z = atomic weight of U, Pu and Zr,

1000 = conversion factor from J/gm to J/kg.

The method of interpolation to evaluate the enthalpy of a U-Pu-Zr fuel composition at and below its solidus temperature consists of first determining the region of the ternary composition triangle (shown in Figure 10.3.9) in which the desired fuel composition lies, and then finding the weight fractions in which the 3 database alloys of the region are required to be mixed to obtain the desired fuel composition. The mass balance of he 3 constituents requires that the following 3 equations be satisfied by the weight fractions of the 3 database alloys.

$$W_{u1}X_1 + W_{u2}X_2 + W_{u3}X_3 = W_u \tag{10.3-10}$$

$$W_{p1}X_1 + W_{p2}X_2 + W_{p3}X_3 = W_p \tag{10.3-11}$$

$$W_{z1}X_1 + W_{z2}X_2 + W_{z3}X_3 = W_z \tag{10.3-12}$$

where

W_{u1}, W_{p1}, W_{z1} = weight fractions of U, Pu and Zr in database alloy 1 of the region in which the desired fuel composition lies,

W_{u2}, W_{p2}, W_{z2} = weight fractions of U, Pu and Zr is database alloy 2 of the region,

W_{u3}, W_{p3}, W_{z3} = weight fractions of U, Pu and Zr in database alloy 3 of the region,

W_u, W_p, W_z = weight fractions of U, Pu and Zr in the desired fuel composition,

X_1, X_2, X_3 = weight fractions of database alloys 1, 2, and 3 in which they must be mixed.

It should be noted that the sum of U, Pu and Zr weight fractions in each database alloy is 1.

$$W_{u,i} + W_{p,i} + W_{z,i} = 1, \quad i = 1, 2, 3 \tag{10.3-13}$$

Adding Eq. (10.3-10) to Eq. (10.3-12), and substituting Eq. (10.3-13), one obtains

$$X_1 + X_2 + X_3 = 1 \tag{10.3-14}$$

Eq. (10.3-11), Eq. (10.3-12) and Eq. (10.3-14) provide a set of 3 linear equations in the variables X_1, X_2 and X_3. The solution of this set is given by

$$X_i = \frac{D_i}{D}, \quad i = 1, 2, 3 \tag{10.3-15}$$

where

$$D = (W_{z1} - W_{z2})(W_{p3} - W_{p1}) - (W_{p1} - W_{p2})(W_{z3} - W_{z1}) \tag{10.3-16}$$

$$D_1 = (W_z - W_{z2})(W_{p3} - W_p) - (W_p - W_{p2})(W_{z3} - W_z) \tag{10.3-17}$$

$$D_2 = (W_{z1} - W_z)(W_{p3} - W_{p1}) - (W_{p1} - W_p)(W_{z3} - W_{z1}) \tag{10.3-18}$$

$$D_3 = (W_{z1} - W_{z2})(W_p - W_{p1}) - (W_{p1} - W_{p2})(W_z - W_{z1}) \tag{10.3-19}$$

The sum of X_1, X_2 and X_3 is always 1. In case of interpolation, the values of X_1, X_2 and X_3 are always positive. If any extrapolation were involved, the value of at least one weight fraction becomes negative which is entirely avoided in the present method. For region 4 of Figure 10.3.9, the mixing fractions of the database alloys given by Eq. (10.3-15) to Eq. (10.3-19) simplify to the following:

$$X_2 = \frac{W_z}{W_{z2}} \tag{10.3-20}$$

$$X_3 = W_p \tag{10.3-21}$$

$$X_1 = 1 - X_2 - X_3 \tag{10.3-22}$$

The Zr weight fraction of database alloy 2 to be used in region 1 of Figure 10.3.9 is found by writing the general equation of the line joining points A and F, i.e., the points representing the U-15Pu-10Zr database and the desired fuel composition.

$$W'_z = W_{z1} + \frac{(W_z - W_{z1})}{(W_p - W_{p1})}(W'_p - W_{p1}) \tag{10.3-23}$$

where

W'_p, W'_z = weight fractions of Pu and Zr corresponding to any point on the line AF.

The Zr weight fraction of database alloy 2 is then found by setting $W'_{\mathrm{p}} = 0$ in Eq. (10.3-23) because database 2 is a U-Zr binary alloy.

$$x = W_{\mathrm{z1}} - \frac{(W_{\mathrm{z}} - W_{\mathrm{z1}})}{(W_{\mathrm{p}} - W_{\mathrm{p1}})} W_{\mathrm{p1}} \tag{10.3-24}$$

where

x = Zr weight fraction of database alloy 2 to be used in region 1.

The enthalpy of the desired U-Pu-Zr fuel composition at and below its solidus temperature is then evaluated by mass-weighted averaging of the enthalpies of the 3 database alloys of the region in which the desired fuel composition lies.

$$h\,(T) = 1000 \left(X_1 \frac{H_1}{a_1} + X_2 \frac{H_2}{a_2} + X_3 \frac{H_3}{a_3} \right) \tag{10.3-25}$$

where

$h\,(T)$ = enthalpy (J/kg) of the U-Pu-Zr fuel composition at a temperature T, at and below its solidus,

H_1, H_2, H_3 = enthalpies (J/mol) of the 3 database alloys at temperature T,

a_1, a_2, a_3 = average atomic weights of the 3 database alloys,

1000 = conversion factor from J/gm to J/kg.

10.3.2.1.5 Subroutine HFUEL1 and Specific Heat by Fuel Type

Figure 10.3.17 shows flow diagram of the subroutine HFUEL1 for calculating the enthalpy, based on this method of regionwise interpolation, of a U-Pu-Zr alloy fuel as a function of composition and temperature. The specific heat of each input fuel type of U-Pu-Zr alloy is evaluated using the 9 parameters defined in Figure 10.3.18 that are calculated using the subroutine HFUEL1 as described in this section. Figure 10.3.19 to Figure 10.3.22 show the comparison of calculated enthalpy with all the data in the Metallic Fuels Handbook for U-15Pu-10Zr and U-10Zr alloys, and U and Pu metals. Figure 10.3.23 to Figure 10.3.25 show the variation of U-Pu-Zr alloy fuel enthalpy with composition obtained from this subroutine. The Zr content varies from 2 to 20 wt % and the Pu content varies from 5 to 30 wt % in the U-Pu-Zr alloy fuels plotted in these figures. The solidus and liquidus temperature required as input to the HFUEL1 subroutine were calculated using the subroutine developed by Pelton [10-13].

The subroutine is used in the SAS4A/SASSYS-1 codes by the subroutine PRECAl (see flow diagram of Figure 10.3.7), before the steady state calculation, to pre-calculate coefficient of a simplified specific heat equation to be used in the steady state and transient calculations of the codes. These coefficients or parameters are defined in Figure 10.3.18, and are pre-calculated for each fuel type that use the regionwise interpolated U-Pu-Zr thermal properties (input option IFUELM=0). This pre-calculation of the coefficients or parameters of the simplified specific heat equation should increase the computational speed of the codes. For this pre-calculation, a sufficient number of pairs of temperature and enthalpy values for the pre-transition solid U-Pu-Zr alloy (see Figure 10.3.18) are obtained from the subroutine HFUEL1. The following enthalpy equation, basically a quadratic function of temperature that is constrained to exactly reproduce the first and last pairs of values, is then fitted by the method of least squares.

$$h\,(T) = f\,(T) + c\,(T - T_1)\,(T - T_{\mathrm{N}}) \tag{10.3-26}$$

$$f\,(T) = h_1 + \frac{h_{\mathrm{N}} - h_1}{T_{\mathrm{N}} - T_1}\,(T - T_1) \tag{10.3-27}$$

where

T_i, h_i = ith pair of temperature and enthalpy values,

N = number of pairs of values,

c = coefficient of the quadratic term in the simplified enthalpy equation, to be determines by the method of least squares.

Minimizing the sum of squared errors, $\sum_{i=1}^{N} \left[h\left(T_i\right) - h_i\right]^2$, one obtains the following value of the quadratic term coefficient.

$$c = \frac{\sum_i \left[h_i - f\left(T_i\right)\right]\left(T_i - T_1\right)\left(T_i - T_N\right)}{\sum_i \left(T_i - T_1\right)^2 \left(T_i - T_N\right)^2} \tag{10.3-28}$$

Differentiating Eq. (10.3-26) with respect to temperature, one obtains the following values of the coefficients of the simplified specific heat equation for the pre-transition solid U-Pu-Zr alloy (see Figure 10.3.18).

$$C_p = \text{ACPF1} + \text{ACPF2} * T \tag{10.3-29}$$

$$\text{ACPF1} = \frac{\left(h_N - h_i\right)}{\left(T_N - T_1\right)} - c\left(T_1 + T_N\right) \tag{10.3-30}$$

$$\text{ACPF2} = 2C \tag{10.3-31}$$

The least-squares fitting procedure is repeated for the post-transition solid U-Pu-Zr alloy (see Figure 10.3.18) to obtain the coefficients its simplified specific heat equation. In each of the remaining temperature ranges of Figure 10.3.18, the specific heat is assumed to be a constant.

It is important to note that, in order to be consistent with the method of Section 3.3.5 for adjusting fuel temperature (computed without accounting for heat of fusion during the main heat transfer calculation) to account for heat of fusion at any radial mesh interval that is going through the melting range, the actual specific heat ACPF6 (see Figure 10.3.18) over the melting range is replaced in subroutine CFUEL by a value linearly interpolated between (i) the post-transition solid U-Pu-Zr specific heat at the solidus temperature, and (ii) the molten liquid solution specific heat, ACPF7, at the liquidus temperature.

This approach of calculating U-Pu-Zr fuel specific heat by fuel type is used only in case of a single radial fuel zone (IFUELC=0), or multiple (i.e., up to 3) radial fuel zones (IFUELC=1). When each fuel radial mesh interval is a unique zone of a different alloy composition (IFUELC=2), then use is made of another subroutine CFUEL1 that calculates U-Pu-Zr alloy specific heat as a function of composition, directly by regionwise interpolation of the temperature-derivative of enthalpies of database alloys, as described in the next section.

10.3.2.2 Specific Heat of U-Pu-Zr Fuel

When each fuel radial mesh interval is a unique zone of a different alloy composition (input parameter IFUELC = 2), then it is difficult to use specific heat by fuel type because only 8 fuel types are currently allowed in the SAS4A/SASSYS-1 codes. The situation becomes even more difficult after in-pin fuel melting, and mixing of fuel radial mesh intervals in the molten fuel cavity produces new alloy compositions. In this case, use is made of another subroutine CFUEL1 that calculates U-Pu-Zr alloy specific heat as a function of composition, directly by regionwise interpolation of the temperature-derivatives of enthalpies of database alloys, given by Eq. (10.3-2) to Eq. (10.3-8). The regionwise interpolation of the derivatives in subroutine CFUEL1 is performed following exactly the same logic as that used in the regionwise interpolation of enthalpies in the subroutine HFUEL1. This makes the two subroutines consistent. Figure 10.3.26 shows flow diagram of the subroutine CFUEL1.

The calculation begins with the evaluation of the first and last solid-state transition temperatures based on data in Table 10.3.1. The bold-boundary rectangular box in the flow diagram of Figure 10.3.26 represents the most major part of the subroutine CFUEL1, the part that performs regionwise interpolation of the temperature-derivatives of enthalpies of database alloys in the temperature range of pre-transition and post-transition solid

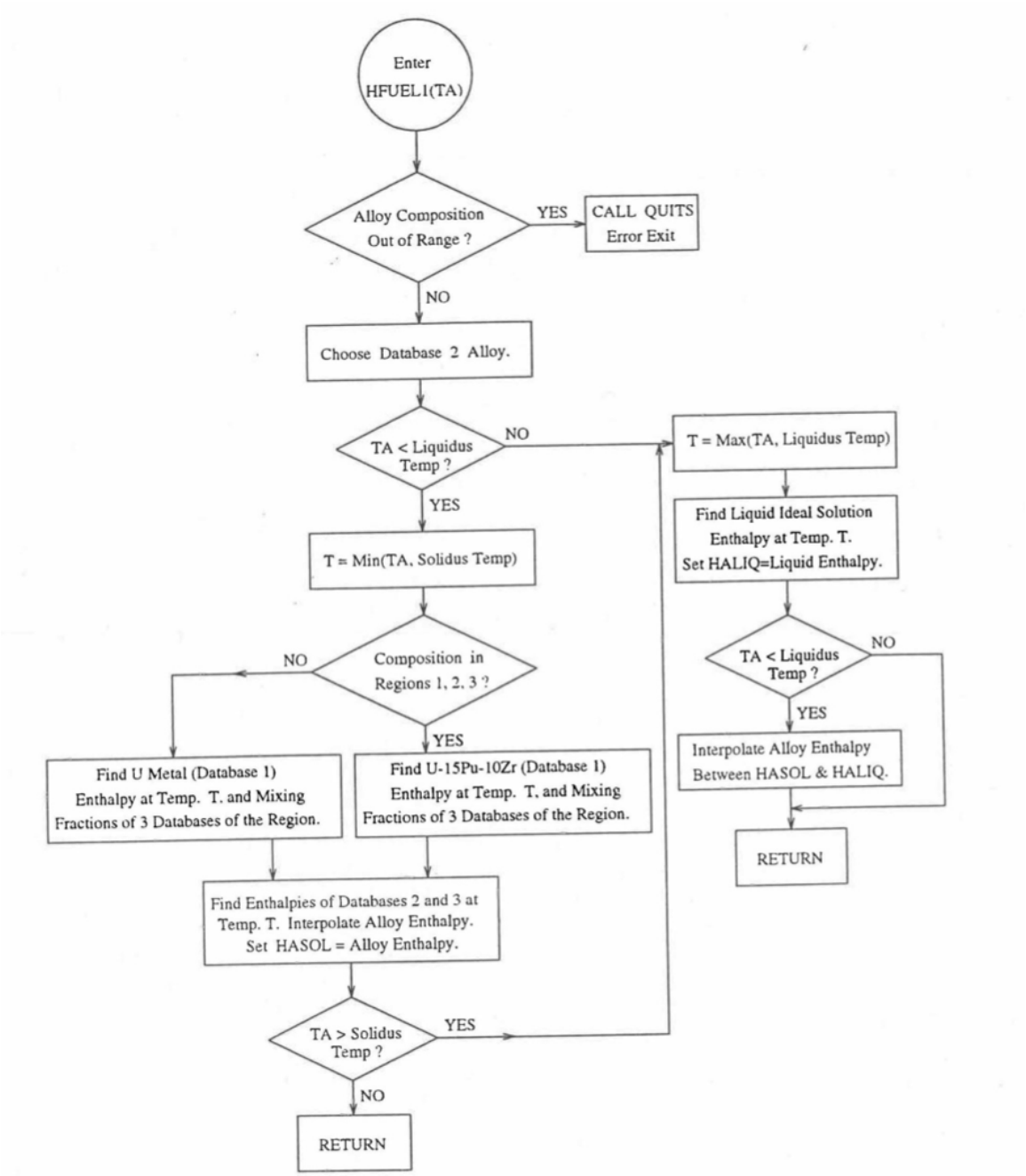


Figure 10.3.17: Flow Diagram of Subroutine HFUEL1 for U-Pu-Zr Enthalpy.

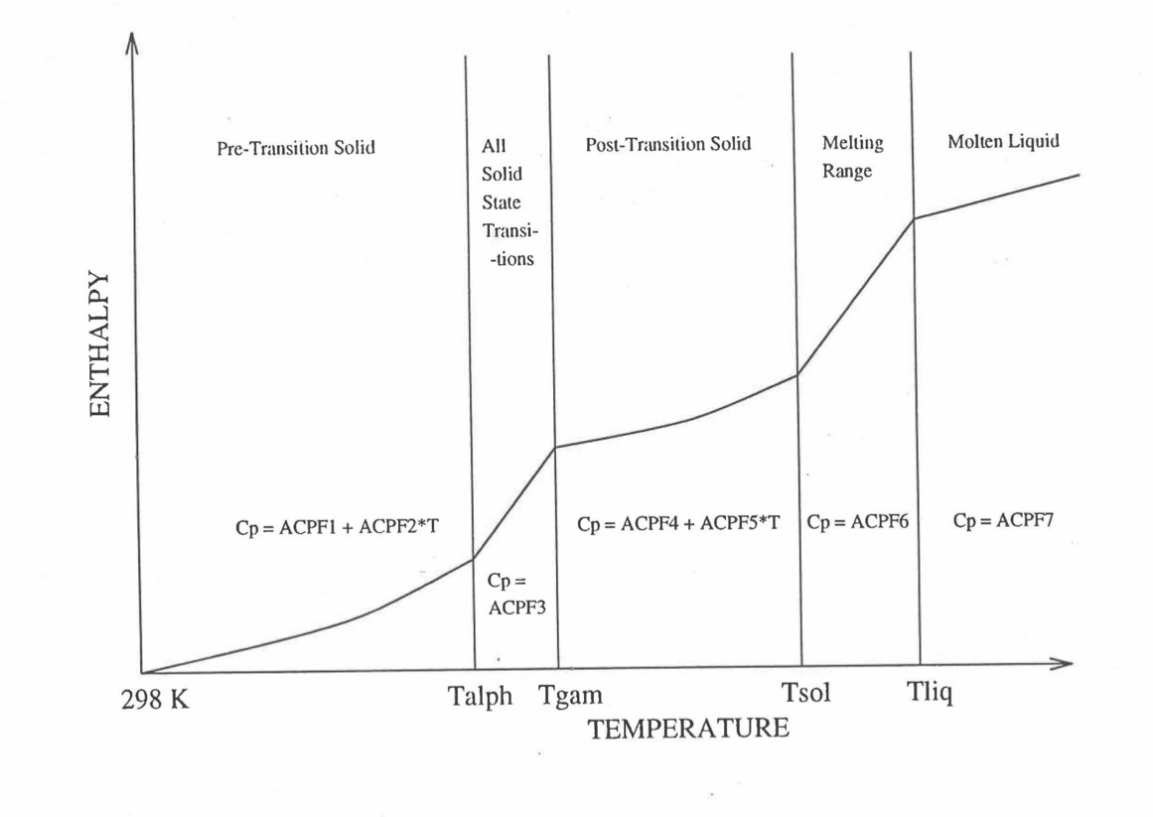


Figure 10.3.18: Diagram Showing the 9 Parameters (ACPF1 to ACPF7, and Solidus and Liquidus Temperatures) Pre-Calculated and Stored for Each U-PuZr Fuel Type for Fast Evaluation of Specific Heat.

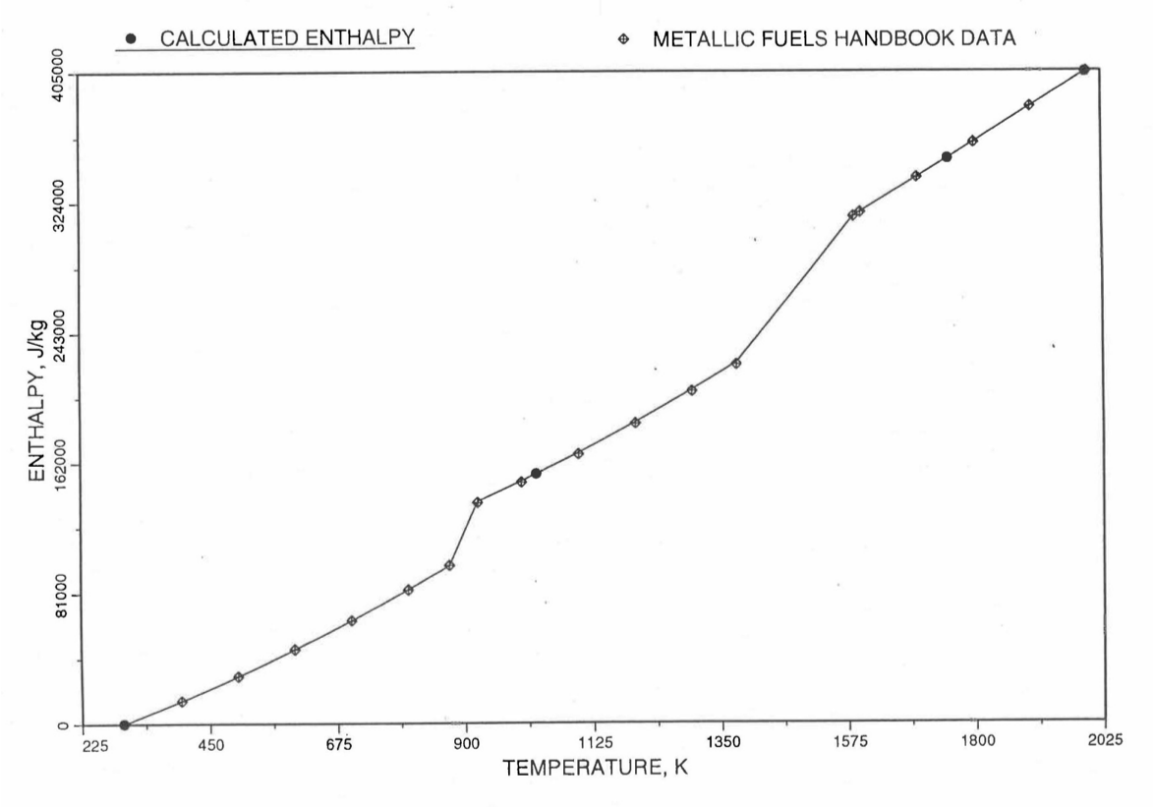


Figure 10.3.19: Comparison of Calculated Enthalpy of U-Pu-10Zr Fuel With the IFR Metallic Fuels Handbook Data.

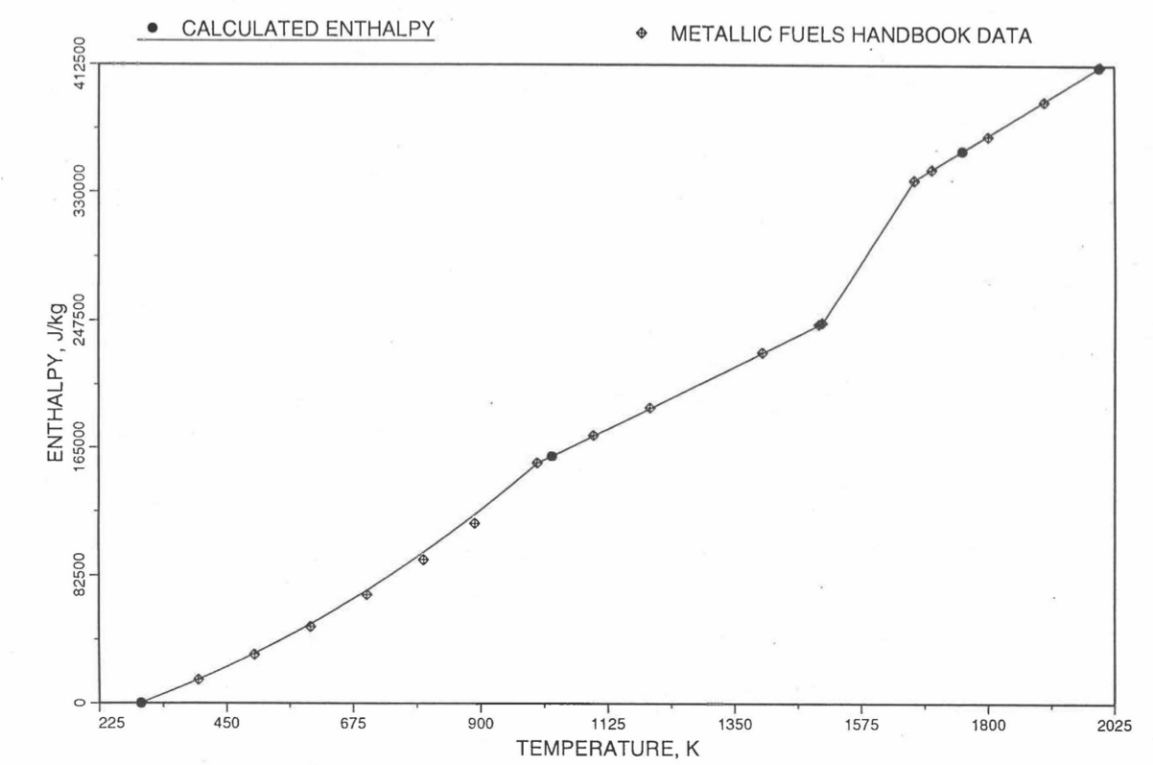


Figure 10.3.20: Comparison of Calculated Enthalpy of U-10Zr Fuel With the Handbook Data (Input KBASE to Subroutine HFUEL1-1).

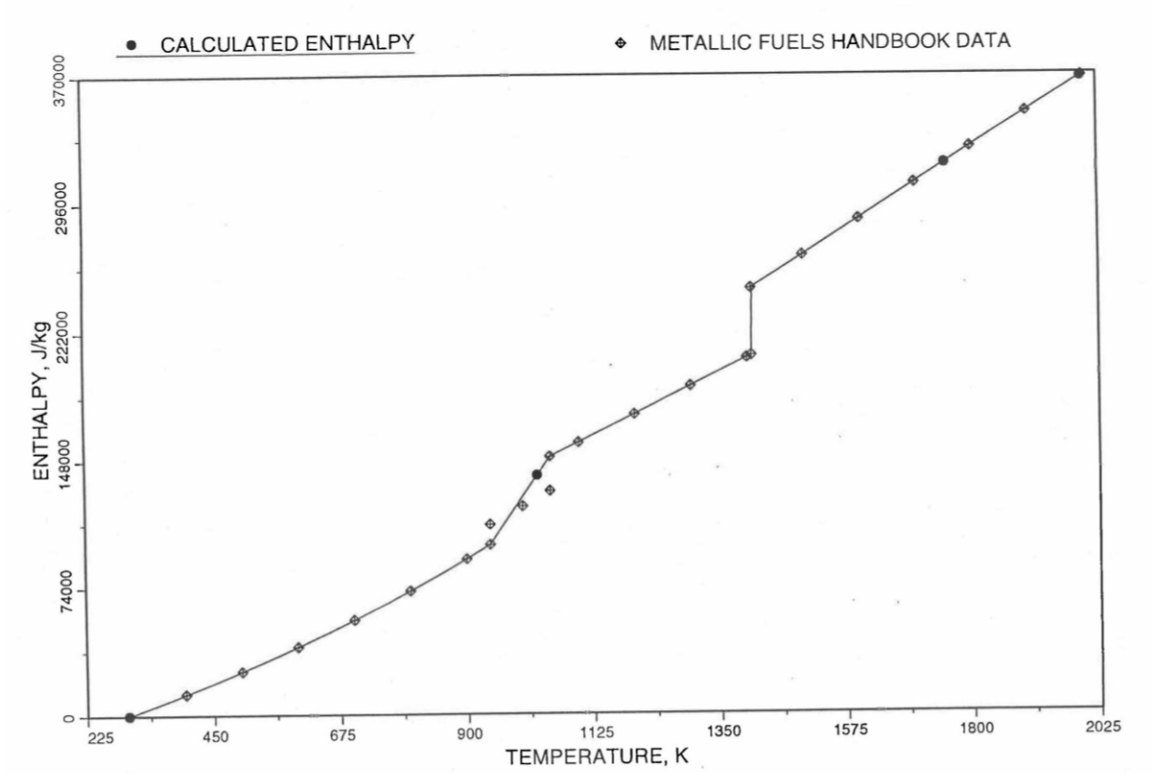


Figure 10.3.21: Comparison of Calculated Enthalpy of Uranium Metal with the IFR Metallic Fuels Handbook Data.

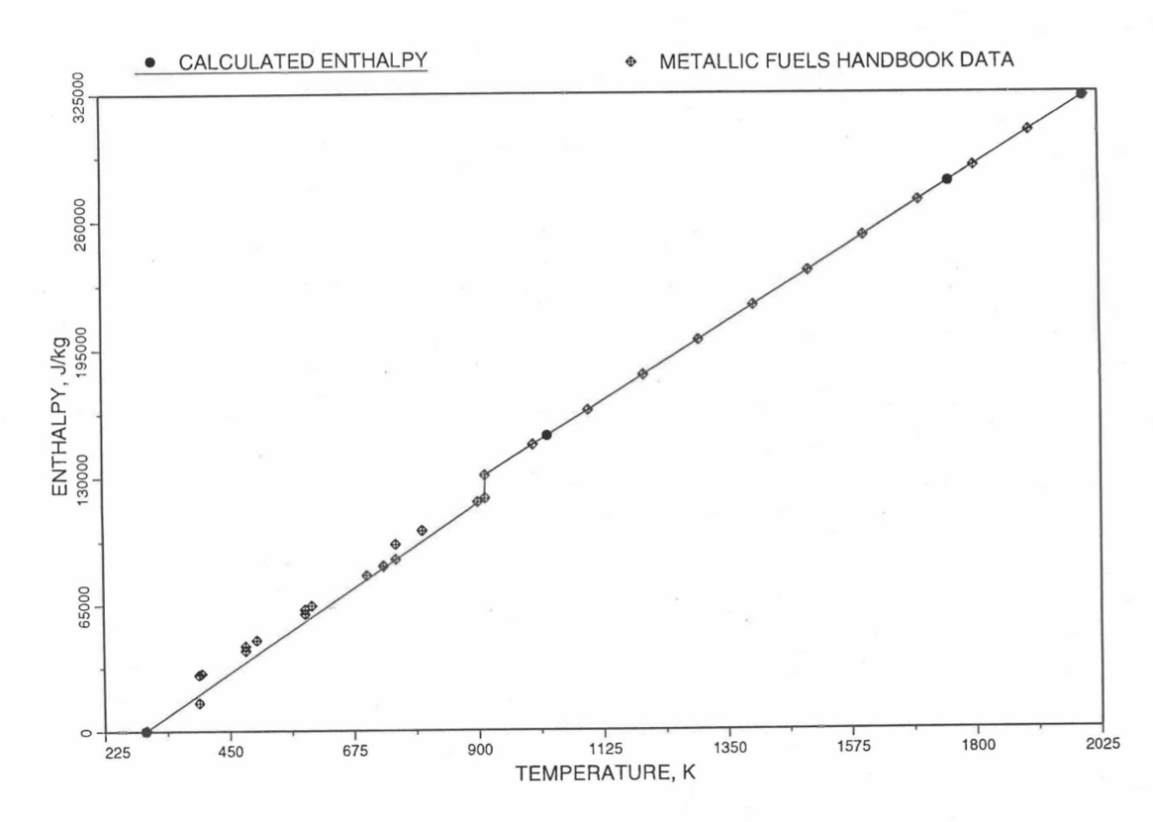


Figure 10.3.22: Comparison of Calculated Enthalpy of Plutonium Metal with the IFR Metallic Fuels Handbook Data.

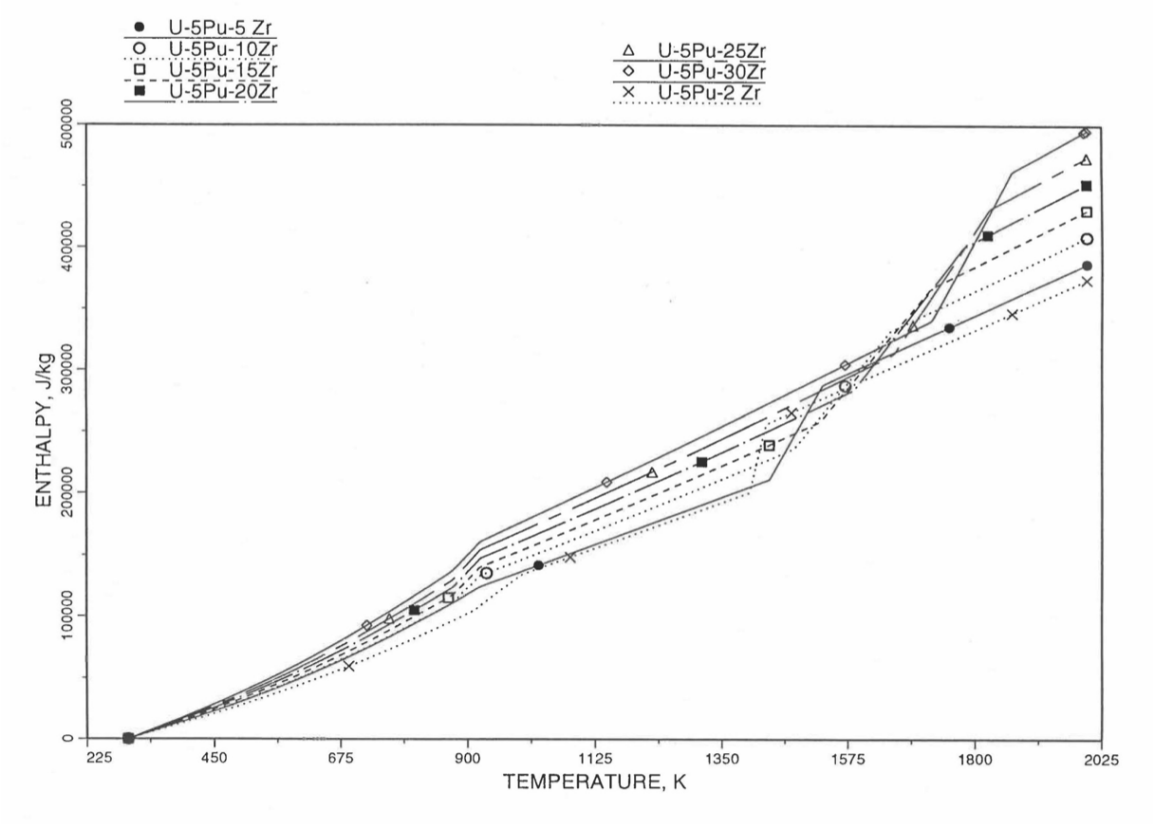


Figure 10.3.23: Variation of U-5Pu-Zr Alloy Fuel Enthalpy With Composition Obtained From Subroutine HFUEL1 Based on the Method of Regionwise Interpolation.

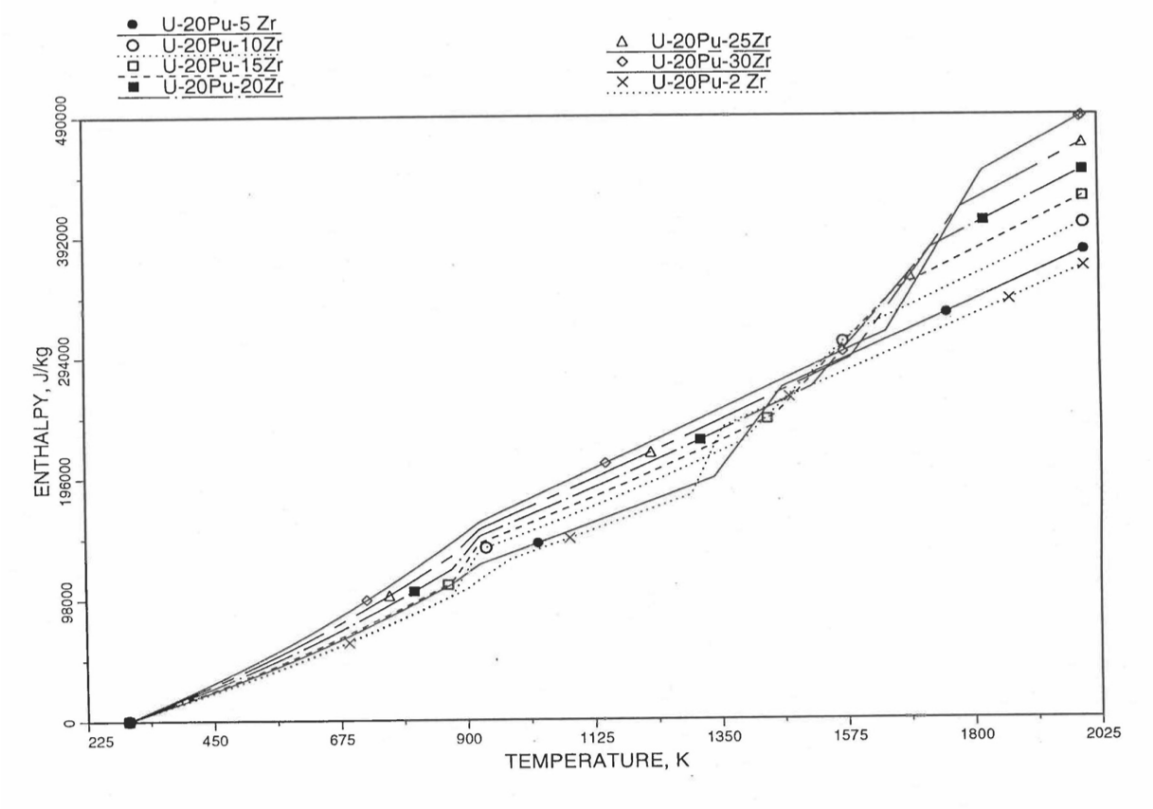


Figure 10.3.24: Variation of U-20Pu-Zr Alloy Fuel Enthalpy With Composition Obtained From Subroutine HFUEL1 Based on the Method of Regionwise Interpolation.

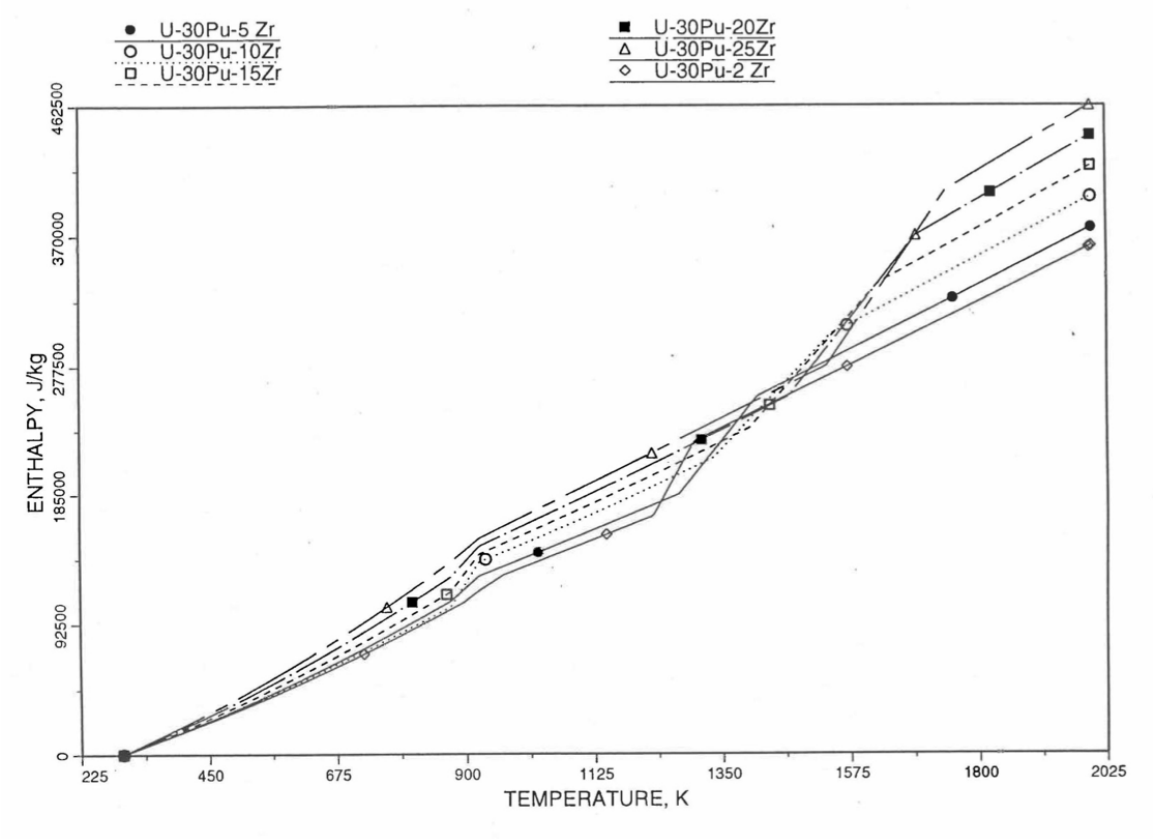


Figure 10.3.25: Variation of U-30Pu-Zr Alloy Fuel Enthalpy With Composition Obtained From Subroutine HFUEL1 Based on the Method of Regionwise Interpolation.

fuels. These temperature ranges are defined in Figure 10.3.18. Between the first and last transition temperatures, the subroutine HFUEL1 is used to find the enthalpies at the first and last transition temperatures, and the fuel specific heat is set equal to the average slope of enthalpy between these two temperatures. The detailed variation of specific heat between the first and last transition temperatures was studied using a modified version of CFUEL1 and found to be small (see Figure 10.3.27 to Figure 10.3.29) and inappropriate for inclusion in the heat transfer calculation of the SAS4A/SASSYS-1 codes. In that modified version of CFUEL1, the regionwise interpolation of derivatives represented by the bold-boundary rectangular box of Figure 10.3.26 was used instead of the average slope. At and above the liquidus temperature of the desired U-Pu-Zr alloy, the specific heat as a function of composition is evaluated by assuming the molten alloy to be an ideal solution of the 3 liquid components U, Pu and Zr. Between the solidus and liquidus temperatures of the desired U-Pu-Zr alloy, the actual specific heat (obtained by following the dashed part of the CFUEL1 flow diagram shown in Figure 10.3.26) over the melting range is replaced by a value linearly interpolated between (i) the post-transition solid U-Pu-Zr specific heat (CPSOL in Figure 10.3.26) at the solidus temperature, and (ii) the molten liquid solution specific heat (CPLIQ in Figure 10.3.26) at the solidus temperature. This replacement is required in order to be consistent with the method of Section 3.3.5 for adjusting fuel temperature (computed without accounting for heat of fusion during the main heat transfer calculation) to account for heat of fusion at any radial mesh interval that is going through the melting range.

Figure 10.3.20 shows the variation of Mark-V (U-20Pu-10Zr), U-15Pu-10Zr and U-10Zr fuel specific heats with temperature, obtained from this subroutine. Figure 10.3.31 to Figure 10.3.36 show the variation of U-Pu-Zr alloy fuel specific heat with composition. The Zr content varies from 2 to 25 wt %, and the Pu content varies from 5 to 30 wt % in the U-Pu-Zr alloy fuels plotted in these figures. Figure 10.3.32, Figure 10.3.34 and Figure 10.3.36 simply show an enlargement of the solid-stat transition region of Figure 10.3.31, Figure 10.3.33 and Figure 10.3.35. The solidus and liquidus temperatures required as input to the CFUEL1 subroutine were obtained using the subroutine developed by Pelton [10-13].

10.3.2.3 Theoretical Density of U-Pu-Zr Fuel

10.3.2.3.1 Introduction

The IFR Metallic Fuels Handbook presents density and thermal expansion data for the 3 constituents and certain specific alloys only, and does not present correlations with their coefficients and transition temperatures as functions of composition.

Earlier, before the preparation of the Metallic Fuels Handbook [109], a method was developed to estimate the composition- and temperature-dependent theoretical density of solid U-Pu-Zr alloy fuel [10-4]. In this method, the theoretical density was first evaluated at a fixed temperature of 298K from the theoretical densities of the 3 constituents, i.e., U, Pu and Zr metals, by assuming that there is no volume change due to alloying of the constituents. This U-Pu-Zr theoretical density was adjusted for thermal expansion from 298K to the desired temperature, assuming thermal expansion to be independent of alloy composition and using the U-15Pu-10ZR alloy thermal expansion data for all alloy compositions. A drawback of this method [10-4] of estimating U-Pu-Zr theoretical density is that the dependence of thermal expansion on alloy composition is left unaccounted. Another drawback is that the data for the theoretical densities of the constituent metals and the thermal expansion of U-15Pu-10Zr were obtained from sources [10-14, 10-15] other than the Metallic Fuels Handbook.

A similar method is used in the FPIN2 code [10-11] to estimate the composition- and temperature-dependent theoretical density of U-Pu-Zr alloy fuel. The theoretical density is first evaluated at a fixed temperature of 293K from the theoretical density data in the Handbook of the 3 constituent metals, by assuming that there is no volume change due to alloying of the constituents. The U-Pu-Zr density is then adjusted for temperature difference, using a composition-independent thermal expansion data. This data is obtained simply by averaging the thermal expansion data of U-5Fs, U-10Zr and U-15Pu-10Zr. The dependence of thermal expansion on alloy composition is here also left unaccounted.

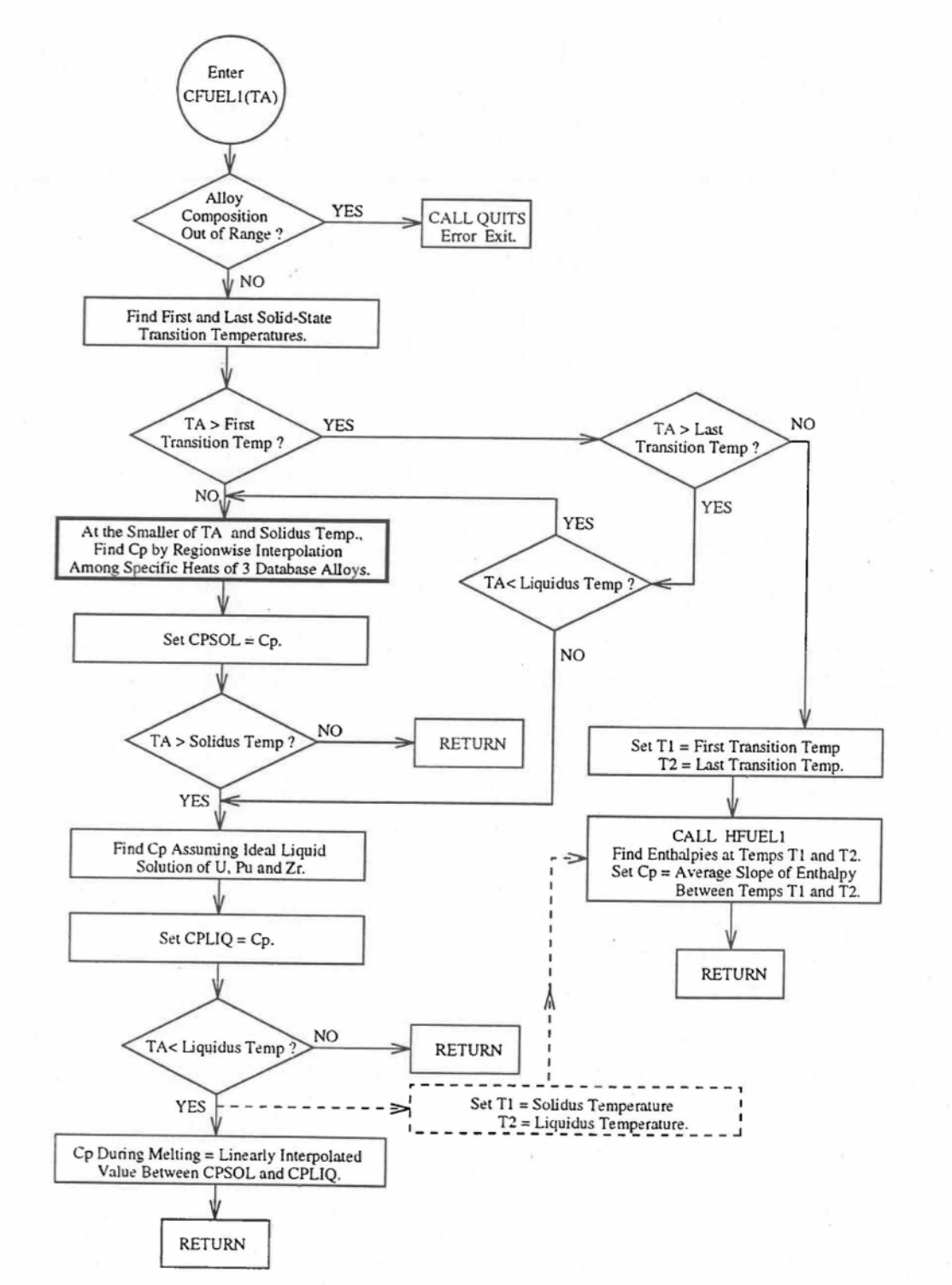


Figure 10.3.26: Flow Diagram of Subroutine CFUEL1 for U-Pu-Zr Specific Heat.

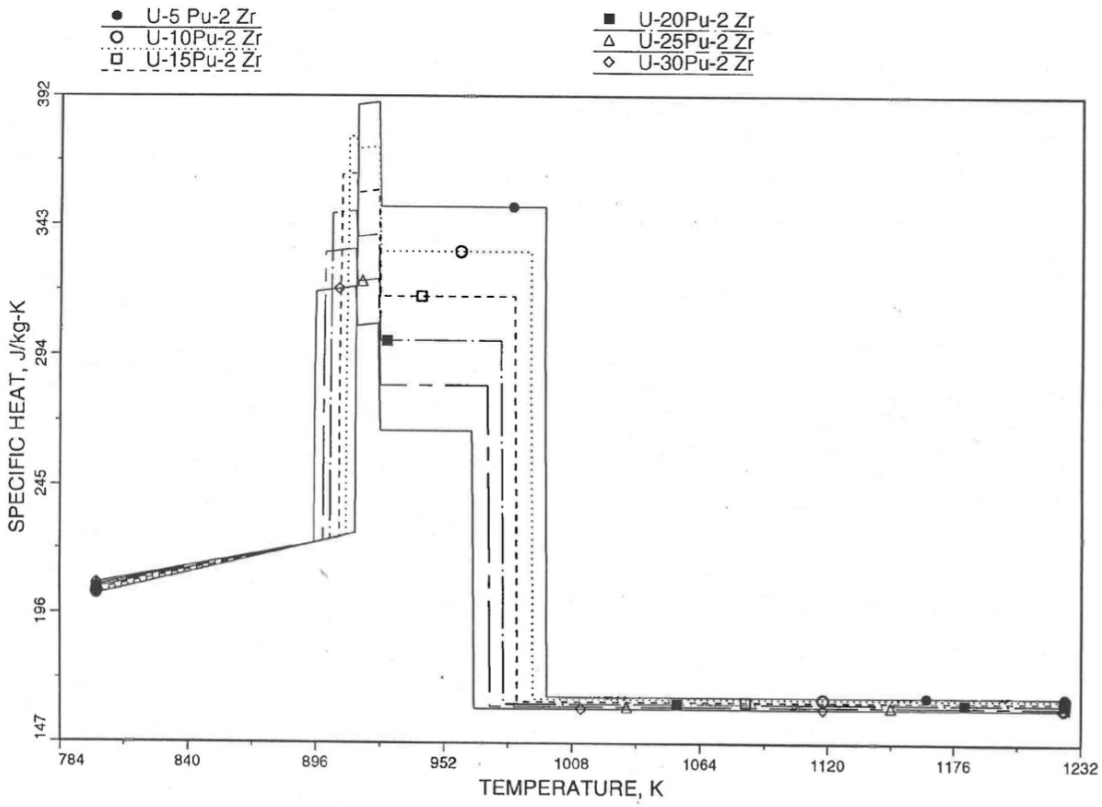


Figure 10.3.27: Detailed Variation of U-Pu-2Zr Fuel Specific Heat Obtained by Regionwise Interpolation of Metallic Fuels Handbook Data.

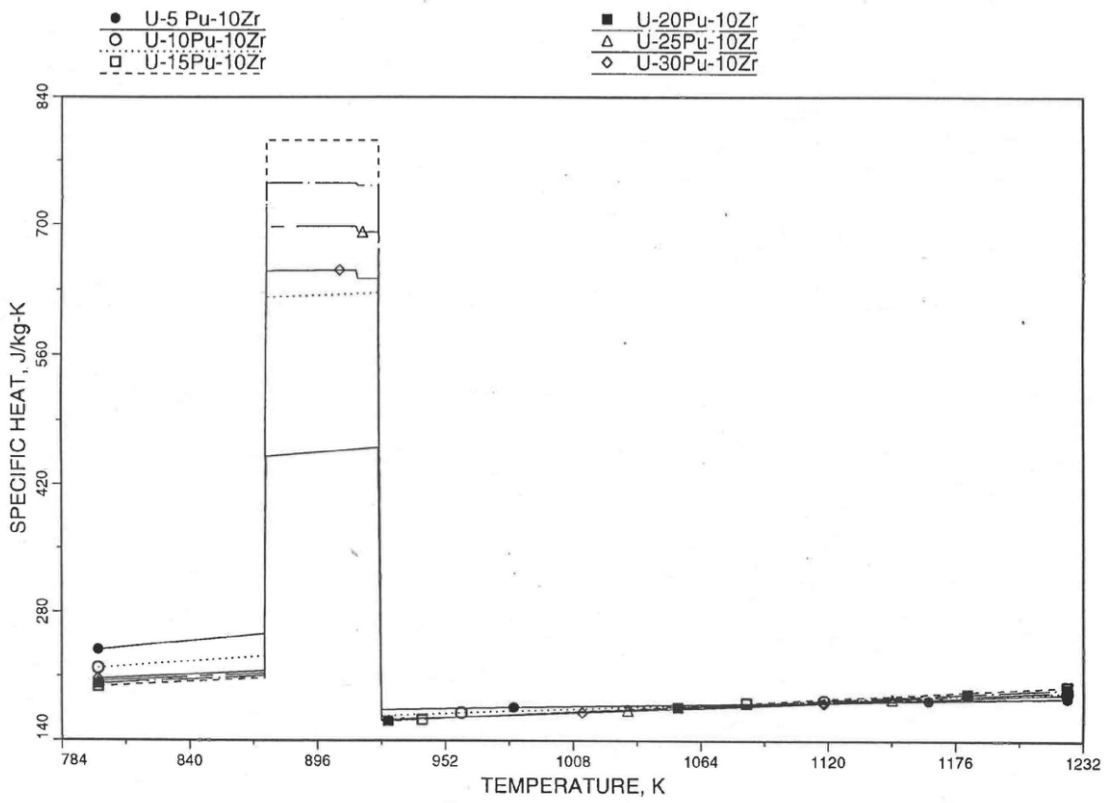


Figure 10.3.28: Detailed Variation of U-Pu-10Zr Fuel Specific Heat Obtained by Regionwise Interpolation of Metallic Fuels Handbook Data.

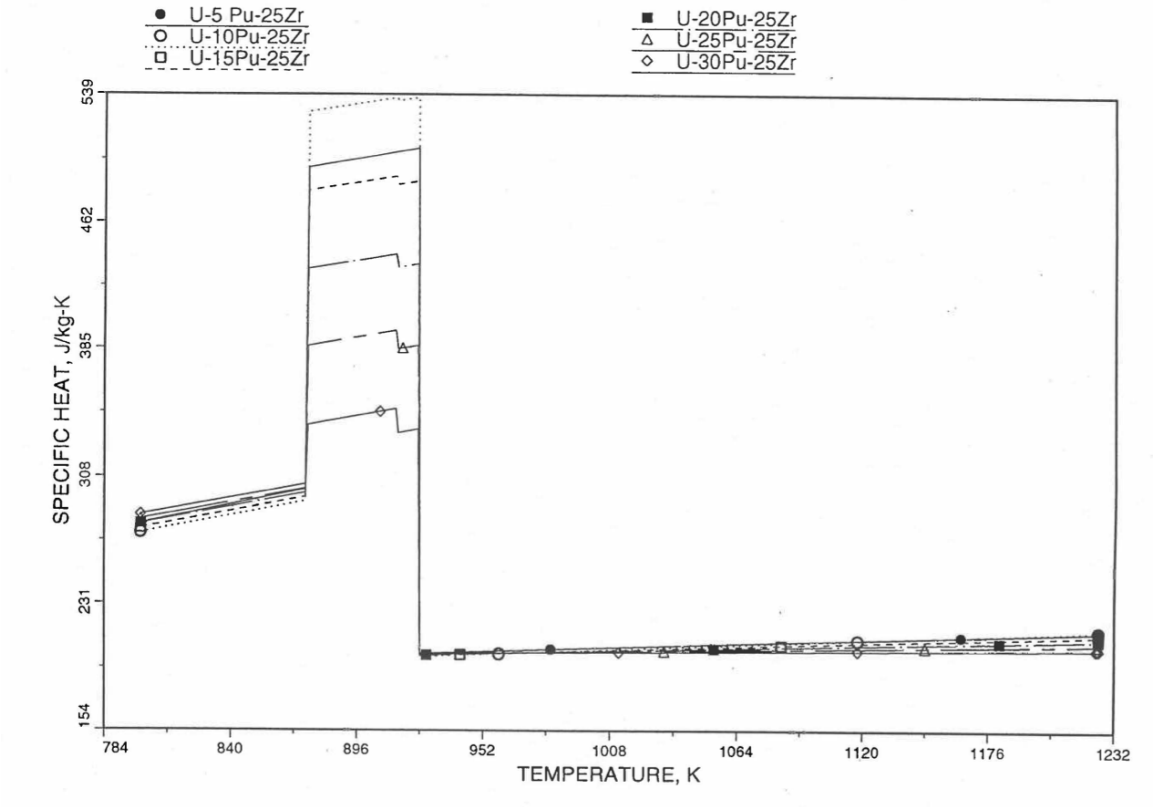


Figure 10.3.29: Detailed Variation of U-Pu-25Zr Fuel Specific Heat Obtained by Regionwise Interpolation of Metallic Fuels Handbook Data.

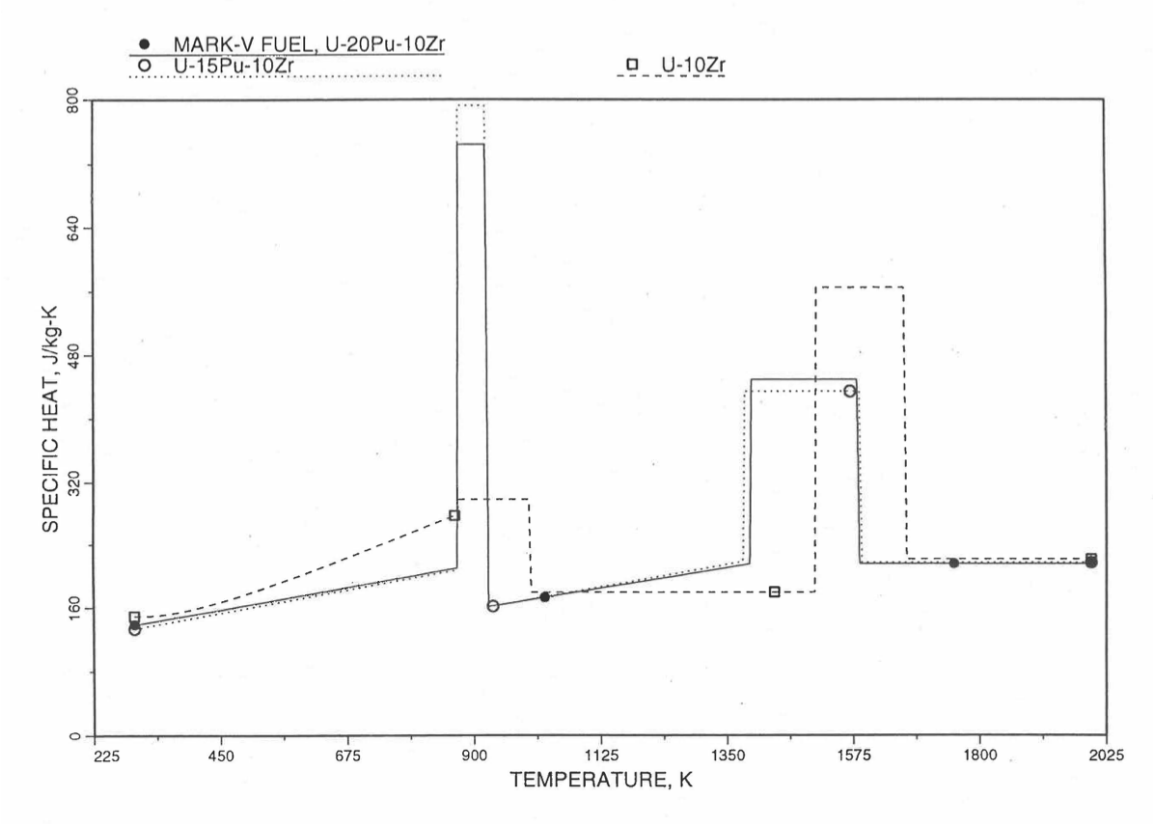


Figure 10.3.30: Specific Heat of Mark-V (U-20Pu-10Zr) Fuel Obtained by Regionwise Interpolation of Metallic Fuels Handbook Data.

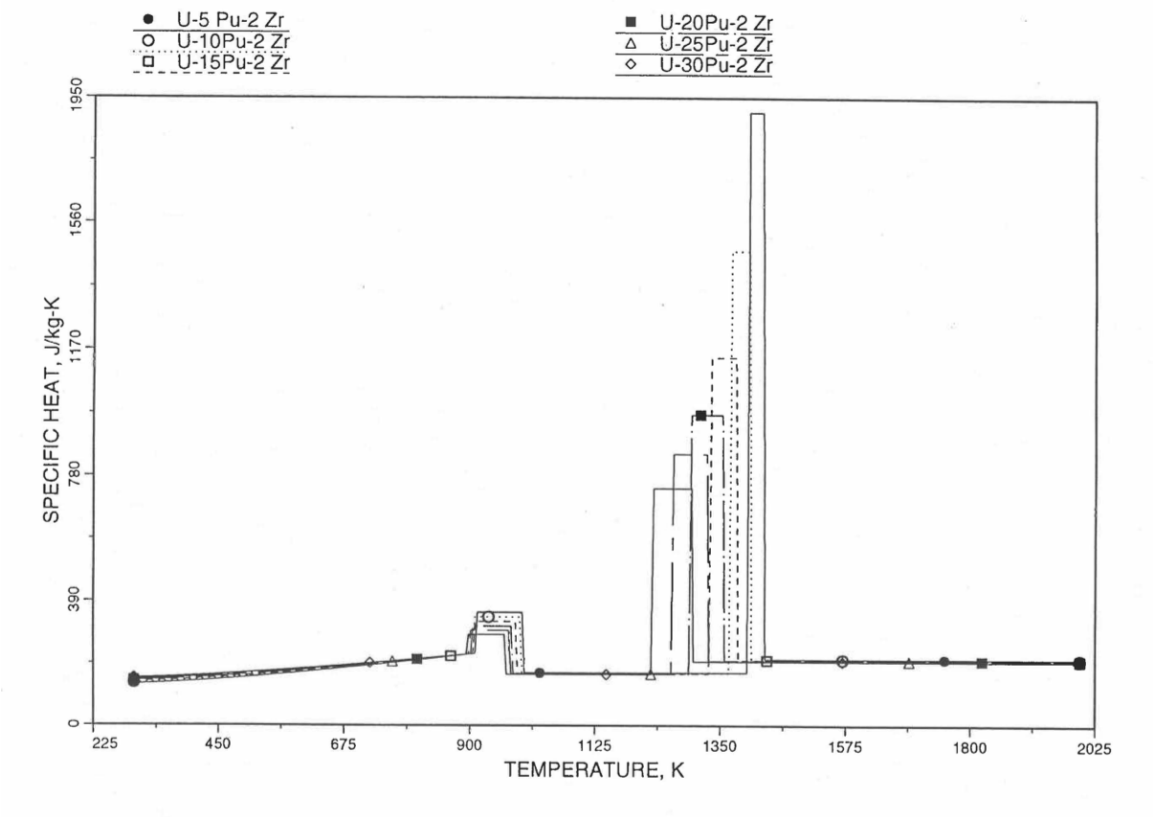


Figure 10.3.31: Specific Heat of U-Pu-2Zr Alloy Fuel Obtained By Regionwise Interpolation of Metallic Fuels Handbook Data.

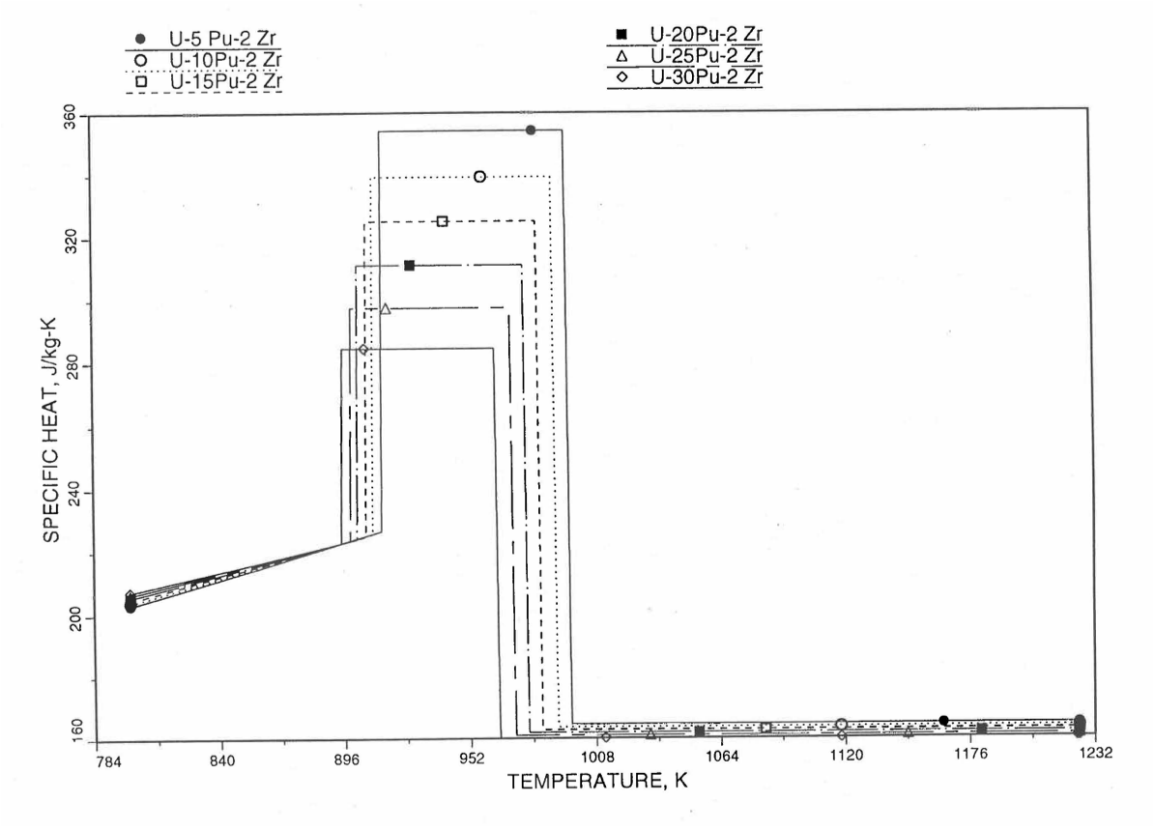


Figure 10.3.32: Specific Heat of U-Pu-2Zr Alloy Fuel Obtained by Regionwise Interpolation of Metallic Fuels Handbook Data (Enlarged).

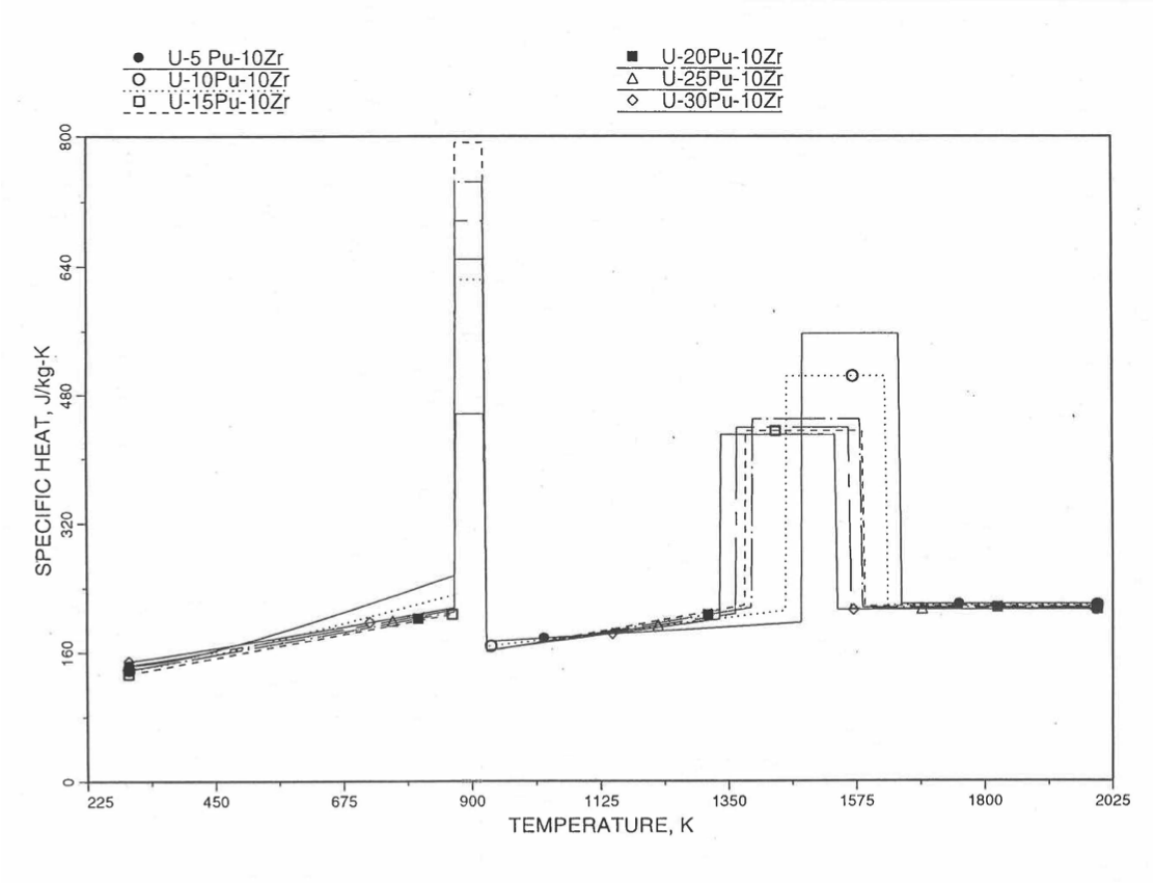


Figure 10.3.33: Specific Heat of U-Pu-10Zr Alloy Fuel Obtained by Regionwise Interpolation of Metallic Fuels Handbook Data.

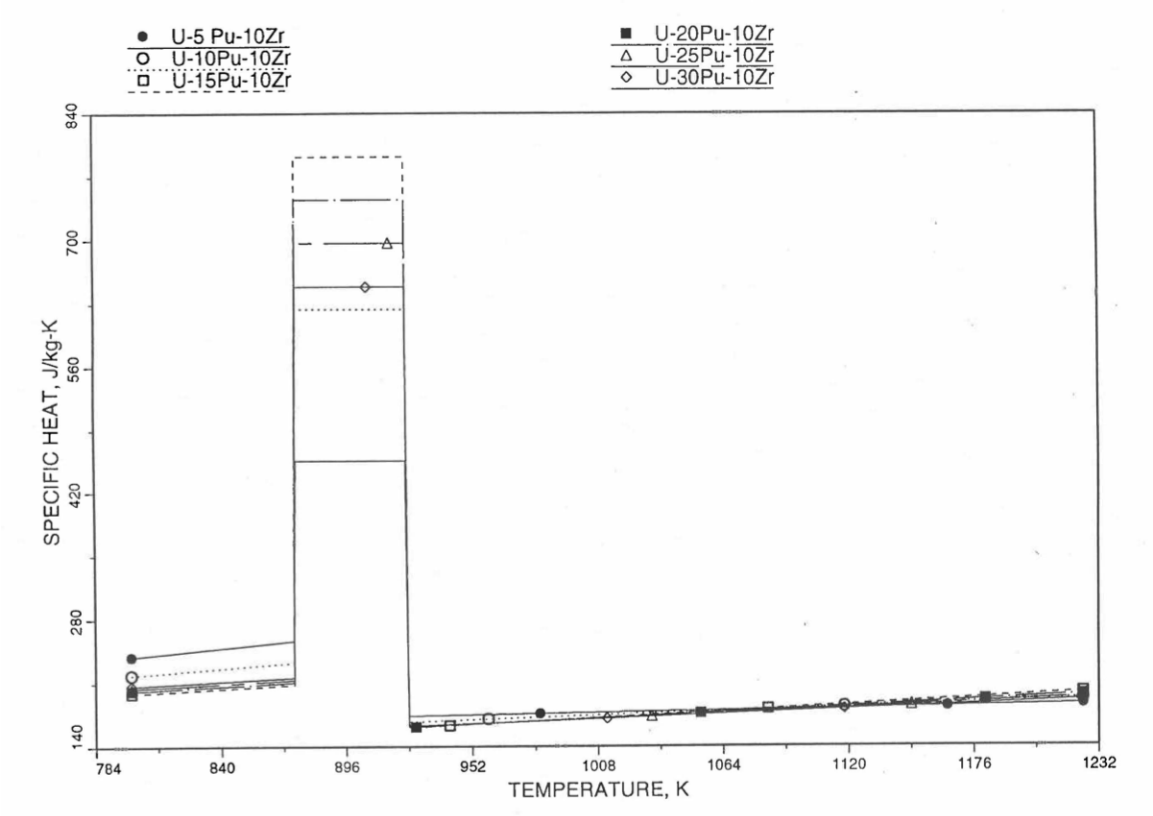


Figure 10.3.34: Specific Heat of U-Pu-10Zr Alloy Fuel Obtained by Regionwise Interpolation of Metallic Fuels Handbook Data (Enlarged).

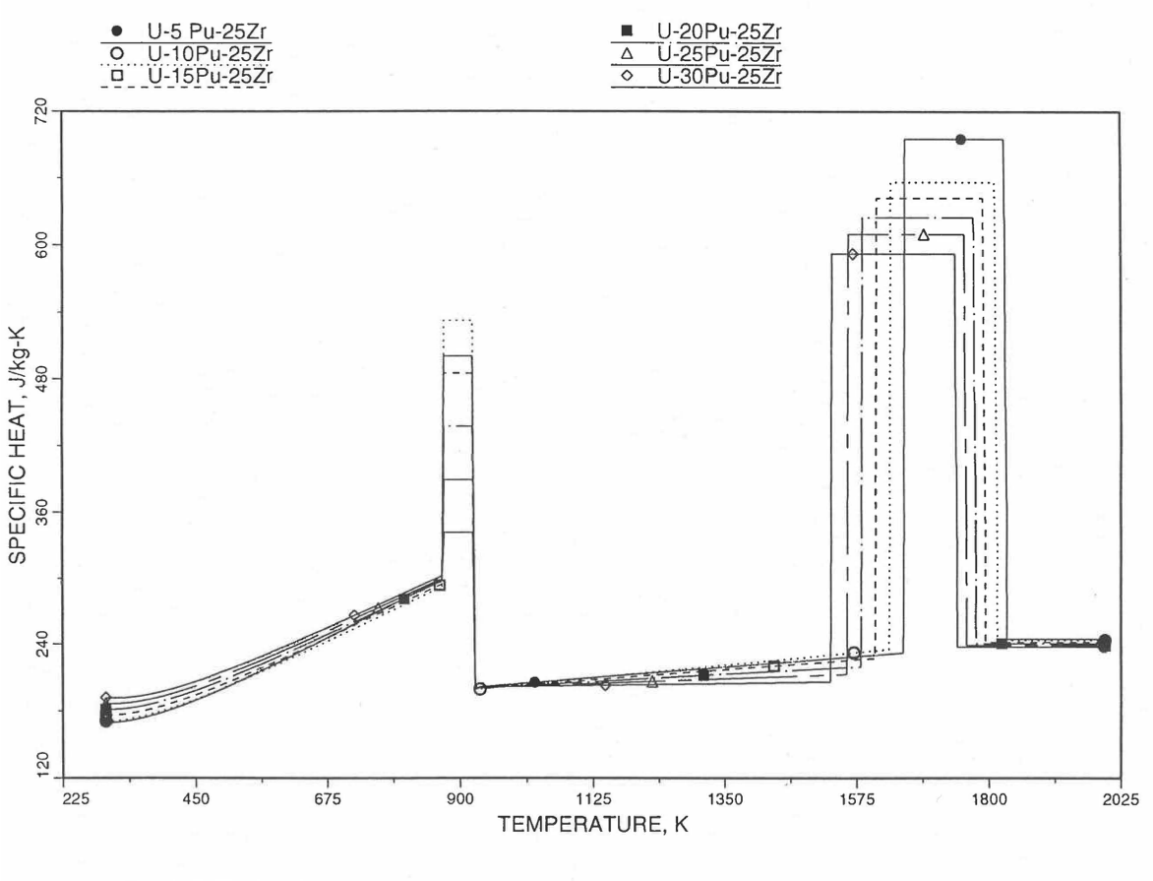


Figure 10.3.35: Specific Heat of U-Pu-25Zr Alloy Fuel Obtained by Regionwise Interpolation of Metallic Fuels Handbook Data.

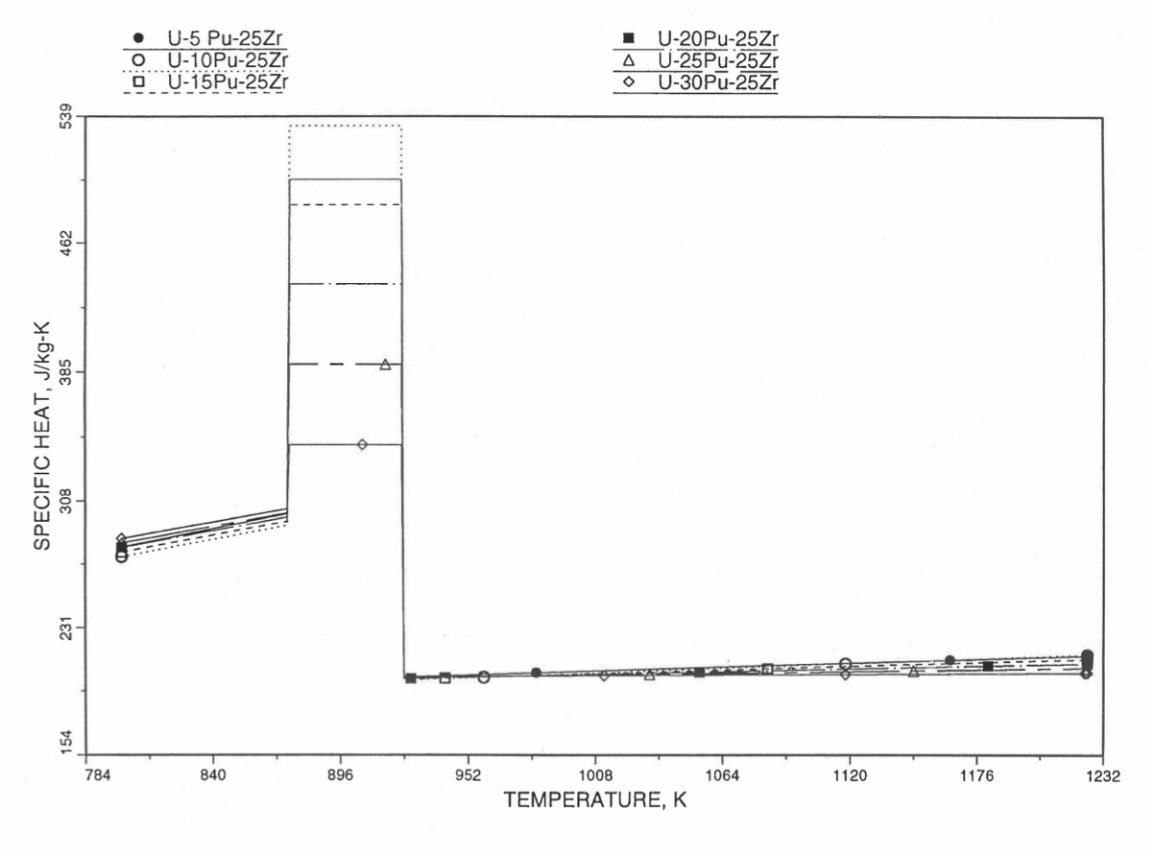


Figure 10.3.36: Specific Heat of U-Pu-25Zr Alloy Fuel Obtained by Regionwise Interpolation of Metallic Fuels Handbook Data (Enlarged).

Recently, new experimental thermal expansion data for U-19Pu-10Zr and U-26Pu-10Zr alloys have been reported in the IFR Handbook [10-8]. Prompted b this new data, the method of estimating U-Pu-Zr alloy enthalpy by regionwise interpolation has been extended to estimate linear thermal expansion of U-Pu-Zr alloy fuels as a function of composition. A subroutine THEXP1 based on this method to calculate the composition-dependent thermal expansion has been developed and incorporated into the SAS4A/SASSYS-1 codes. The subroutine makes use of the thermal expansion data of all the 8 alloys/components reported in the Handbook [10-8]. The 8 basic alloy compositions are used to divide the ternary composition triangle into 9 regions of interpolation as described in Section 10.3.2.4. The method is flexible enough to make use of the thermal expansion data of additional alloys (as they become available) to divide the composition triangle into a larger number of smaller regions, thus improving its accuracy. The U-Pu-Zr theoretical density is now adjusted for temperature difference using composition-dependent thermal expansion calculated for the desired alloy instead of using an arbitrarily assumed composition-independent thermal expansion data. Also, an approximate technique to correct the theoretical density for the presence of fission products, which remains valid even if annular composition zones have been formed due to radial redistribution of alloy constituents, has been developed and incorporated into the SAS4A/SASSYS-1 codes. This technique is applied by a code user option.

10.3.2.3.2 Theoretical Density of U-Pu-Zr Fuel at 293K

The theoretical density of U-Pu-Zr fuel at a fixed temperature of 293K is calculated form the theoretical densities of U, Pu and Zr by assuming that there is no volume change due to alloying of the constituents. This assumption has been referred to as the ideal-solution approximation, and recommended in the IFR Handbook. The recommended theoretical densities of U, Pu and Zr at 293K are 19070, 19750 and 6570 kg/m^3. The volume of one kilogram of U-Pu-Zr alloy can be found by summing the volumes of its constituents, and the reciprocal of this volume equals theoretical density.

$$\rho_{\text{th}}\left(0, 293\text{K}, W_{\text{p}}, W_{\text{z}}\right) = \frac{1}{\left(\frac{W_{\text{u}}}{19070} + \frac{W_{\text{p}}}{19750} + \frac{W_{\text{z}}}{6570}\right)} \tag{10.3-32}$$

where

$\rho_{\text{th}}\left(\text{Bu}, T, W_{\text{p}}, W_{\text{z}}\right)$ = theoretical density of U-Pu-Zr alloy at temperature T°K after a burnup of Bu atom percent, kg/m^3,

$W_{\text{u}}, W_{\text{p}}, W_{\text{z}}$ = weight fractions of U, Pu and Zr in the alloy.

Table 10.3.2 shows a comparison of all the measured densities reported in the IFR Handbook with the theoretical densities calculated using Eq. (10.3-32). The comparison gives a maximum error of 2.45% with an average error of 1.1%. The comparison is satisfactory for these alloy compositions. The small differences may be due to measurement error, small voids left during the alloy preparation or some volume change taking place during alloying. Eq. (10.3-32) does not account for the effect of isotopic compositions of U and Pu on theoretical density, and some of the difference between the measured and calculated densities may be caused by this effect.

10.3.2.3.3 Correction for the Presence of Fission Products

Eq. (10.3-32) gives the theoretical density of unirradiated U-Pu-Zr fuel at a fixed temperature of 293K. This section presents the effect of the presence of fission products, and the next section presents the effect of temperature on U-Pu-Zr fuel theoretical density. The effect of burnup on U-Pu-Zr fuel density was studied earlier [10-16], and the following equation for the theoretical density of irradiated fuel at a fixed temperature of 293K was obtained in terms of fabricated fuel composition and burnup.

Table 10.3.2: Comparison of Measured and Calculated Theoretical Densities of U-Pu-Zr Alloys at 293K

		Density, kg/m^3		
No.	Alloy	Handbook	Calculated Using Eq. (10.3-32)	% Error
1	U	19070	19070	0.0
2	U-10Zr	16020	16022	0.01
3	U-20Zr	13810	13814	0.03
4	Zr	6570	6570	0.0
5	U-15Pu-10Zr	15800	16092	1.85
6	U-19Pu-10Zr	–	16110	–
7	U-26Pu-10Zr	–	16143	–
8	Pu	19750	19750	0.0
9	U-20Pu-10Zr	15730	16115	2.45
10	U-11.1Pu-6.3Zr	16800	17087	1.71
11	U-18.5Pu-14.1Zr	14800	15112	2.11
12	U-15Pu-6.8Zr	16600	16963	2.19
13	U-15Pu-13.5Zr	15000	15235	1.57

$$\rho_{\text{th}}\left(\text{Bu}, 293\text{K}, W'_{\text{p}}, W'_{\text{z}}\right) = \frac{\rho_{\text{th}}\left(0, 293\text{K}, W_{\text{po}}, W_{\text{zo}}\right)\left[1 - 9.026 \times 10^{-6}\text{Bu}\left(W_{\text{uo}} + W_{\text{po}}\right)\right]}{1 + 1.190 \times 10^{-6}\rho_{\text{th}}\left(0, 293K, W_{\text{po}}, W_{\text{zo}}\right)\text{Bu}\left(W_{\text{uo}} + W_{\text{po}}\right)} \tag{10.3-33}$$

where

$W_{\text{uo}}, W_{\text{po}}, W_{\text{zo}}$ = fabricated weight fractions of U, Pu and Zr in the unirradiated U-Pu-Zr alloy,

$W'_{\text{u}}, W'_{\text{p}}, W'_{\text{z}}$ = weight fractions of U, Pu and Zr in the irradiated alloy after a burnup of Bu atom percent, without any radial migration of alloy constituents.

This change in alloy composition is only due to fission and breeding. The value of the fabricated alloy theoretical density required for the evaluation of the right hand side of Eq. (10.3-33) can be obtained from Eq. (10.3-32), using the fabricated weight fractions in place of the present weight fractions of the alloy.

In this section we want to express the irradiated fuel theoretical density in terms of zonal fuel composition (resulting from fission, breeding and also migration of constituents) and burnup. For this purpose, we need to know the amount of fission products the present fuel contains. During irradiation, uranium migrates to the middle radial composition zone while plutonium, zirconium and fission products migrate out of the middle radial composition zone. Thus the fuel constituents and their generated fission products do not remain together. This makes it difficult to calculate the amount of fission products mixed in a given zonal fuel composition. Lacking a modeling and calculation of the radial migration of fission products in a fuel pin, it is assumed that the fission products are uniformly distributed over the pin cross section, on a per unit U-Pu-Zr alloy post-irradiation mass basis. The total amount of fission products is obtained from the radially averaged atom percent burnup and the fabricated pin composition. Due to radial migration of heavy metal constituents, it is difficult to define a radial variation of atom percent burnup. The fabricated composition and not the irradiated (zonal0 composition is the basis of atom percent burnup reported in literature and, therefore, the fabricated composition should be used in evaluating the total amount of fission products. The number of heavy-metal atoms, N_{ho}, per unit fabricated U-Pu-Zr alloy mass is given by

$$N_{\text{ho}} = \frac{W_{\text{uo}}\text{Na}}{a_{\text{u}}} + \frac{W_{\text{po}}\text{Na}}{a_{\text{p}}} \tag{10.3-34}$$

where

a_u, a_p = uranium and plutonium atomic weights,

Na $= 6.02472 \times 10^{26}$ = Avogadro's number, molecules/kg-mole.

The number of fissions, N_{fo}, per unit fabricated mass equals the product of burnup and heavy-metal atoms.

$$N_{fo} = \frac{\text{Na Bu}}{100}\left(\frac{W_{uo}}{a_u} + \frac{W_{po}}{a_p}\right) \tag{10.3-35}$$

Assuming $a_u \approx a_p$, the mass of heavy-metals fissioned, M_{fo}, per unit fabricated U-Pu-Zr alloy mass is given by

$$M_{fo} = N_{fo}\frac{a_u}{\text{Na}} \tag{10.3-36}$$

Substituting Eq. (10.3-35), Eq. (10.3-36) becomes

$$M_{fo} = 0.01\text{Bu}\,(W_{uo} + W_{po}) \tag{10.3-37}$$

The mass of U-Pu-Zr alloy left after irradiation per unit fabricated mass is given by $(1 - M_{fo})$. Therefore, the average number of fissions, N_f, per unit U-Pu-Zr alloy post-irradiation mass is obtained by dividing Eq. (10.3-35) by $(1 - M_{fo})$.

$$N_f = \frac{0.01\text{Bu Na}\frac{(W_{uo}+W_{po})}{a_u}}{1 - 0.01\text{Bu}\,(W_{uo} + W_{po})} \tag{10.3-38}$$

Eq. (10.3-38) determines the assumed uniform distribution of fission products over the pin cross section, on a per unit U-Pu-Zr alloy post-irradiation mass basis. As discussed above, the amount of fission products determined by Eq. (10.3-38) is used to correct U-Pu-Zr alloy theoretical density for the presence of fission products as follows.

Considering the mass-plus-energy conservation of a nuclear power reactor with negligible neutron leakage and capture in coolant and structure, all neutrons emitted by fuel are absorbed in the fuel itself. Therefore, the mass of fission products equals the mass of heavy-metals fissioned minus the mass-equivalent of fission energy released. The mass of fission products, M_f, per unit U-Pu-Zr alloy post-irradiation mass is given by

$$M_f = \frac{N_f a_u}{\text{Na}} - \frac{200 m_a}{931}N_f \tag{10.3-39}$$

where

m_a = 1 atomic mass unit = 1.659828×10^{-27} kilogram,

200 = energy released per fission in MeV units,

931 = energy-equivalent (in MeV units) of 1 atomic mass unit.

Following Ref. [10-16], the volume of fission products, V_f, per unit U-Pu-Zr alloy post-irradiation mass is given by

$$V_f = N_f v_{sfp} \tag{10.3-40}$$

where

v_{sfp} = characteristic volume of the solid fission products per fission = 6.772565×10^{-29} m^3.

This value of v_{sfp} has been found from pure uranium swelling data to be 4.70×10^{-29} m^3/fission over and above the characteristic volume of uranium atom [10-16]. Using Eq. (10.3-39) and Eq. (10.3-40), the theoretical

density of irradiated fuel at a fixed temperature of 293K, after correction for the presence of fission products, is obtained as

$$\rho_{\text{th}}\left(\text{Bu}, 293\text{K}, W_{\text{p}}, W_{\text{z}}\right) = \frac{1 + \frac{N_{\text{f}} a_{\text{u}}}{\text{Na}} - 200 m_{\text{a}} \frac{N_{\text{f}}}{931}}{\left(\frac{W_{\text{u}}}{19070} + \frac{W_{\text{p}}}{19750} + \frac{W_{\text{z}}}{6570}\right) + N_{\text{f}} v_{\text{sfp}}} \tag{10.3-41}$$

where

$W_{\text{u}}, W_{\text{p}}, W_{\text{z}}$ = weight fractions of U, Pu and Zr in the irradiated U-Pu-Zr alloy.

Using a value of 238 for a_{u}, and noting that m_{a} is the reciprocal of Na, Eq. (10.3-41) becomes

$$\rho_{\text{th}}\left(\text{Bu}, 293\text{K}, W_{\text{p}}, W_{\text{z}}\right) = \frac{1 + 0.009991 f_{\text{o}}}{\left(\frac{W_{\text{u}}}{19070} + \frac{W_{\text{p}}}{19750} + \frac{W_{\text{z}}}{6570}\right) + 1.7144 \times 10^{-6} f_{\text{o}}} \tag{10.3-42}$$

where

$$f_{\text{o}} = \frac{\text{Bu}\left(W_{\text{uo}} + W_{\text{po}}\right)}{1 - 0.01\text{Bu}\left(W_{\text{uo}} + W_{\text{po}}\right)} \tag{10.3-43}$$

The quantity f_{o} is the heavy-metal mass fissioned expressed as a percentage of U-Pu-Zr alloy post-irradiation mass. This approximate technique to correct the theoretical density for the presence of fission products, which remains valid even after annular zone formation ($W_{\text{u}}, W_{\text{p}}, W_{\text{z}}$ are then zonal weight fractions) due to radial migration of alloy constituents, is incorporated into the SAS4A/SASSYS-1 codes and is applied by a code user option.

10.3.2.3.4 Theoretical Density of Irradiated U-Pu-Zr Fuel

Eq. (10.3-42) and Eq. (10.3-43) give the theoretical density of irradiated U-Pu-Zr fuel at a fixed temperature of 293K. In the next section, a method of interpolating the linear thermal expansion from 293K to any desired temperature T as a function of fuel composition is described. Using this thermal expansion, the following equation can be written for the theoretical density of irradiated U-Pu-Zr fuel at any desired temperature T:

$$\rho_{\text{th}}\left(\text{Bu}, T, W_{\text{p}}, W_{\text{z}}\right) = \frac{\left(1 + 0.009991 f_{\text{o}}\right)\left[1 + \Delta L\left(T, W_{\text{p}}, W_{\text{z}}\right)/\text{Lo}\right]^{-3}}{\left(\frac{W_{\text{u}}}{19070} + \frac{W_{\text{p}}}{19750} + \frac{W_{\text{z}}}{6570}\right) + 1.7144 \times 10^{-6} f_{\text{o}}} \tag{10.3-44}$$

where

$\Delta L\left(T, W_{\text{p}}, W_{\text{z}}\right)/L_{\text{o}}$ = linear thermal expansion from 293K to temperature T, expressed as a fraction of length L_{o} at the temperature 293K, for a U-Pu-Zr alloy with U, Pu, and Zr weight fractions equal to $W_{\text{u}}, W_{\text{p}}$ and W_{z}.

Figure 10.3.37 shows the flow diagram of a subroutine RHOFM for calculating the theoretical density, based on this method, of irradiated U-Pu-Zr alloy fuel as a function of composition and temperature. For improving the computational speed of the SAS4A/SASSYS-1 codes, the theoretical density without fission products at 293K of each U-Pu-Zr fuel type (that uses the regionwise interpolation thermal properties, i.e., input option IFUELM = 0) is precalculated in subroutine PRECAL before steady state calculation, and used later in the steady state and transient calculation of the codes.

Figures (Figure 10.3.38) to (Figure 10.3.41) show the variation of theoretical density (without fission products) with temperature for 28 alloy compositions, calculated using Eq. (10.3-44). The thermal expansion required on the right hand side of Eq. (10.3-44) was obtained using the subroutine THEXP1 described below. The Pu and Zr contents each vary from zero to 30 at % in the alloy compositions plotted in these figures.

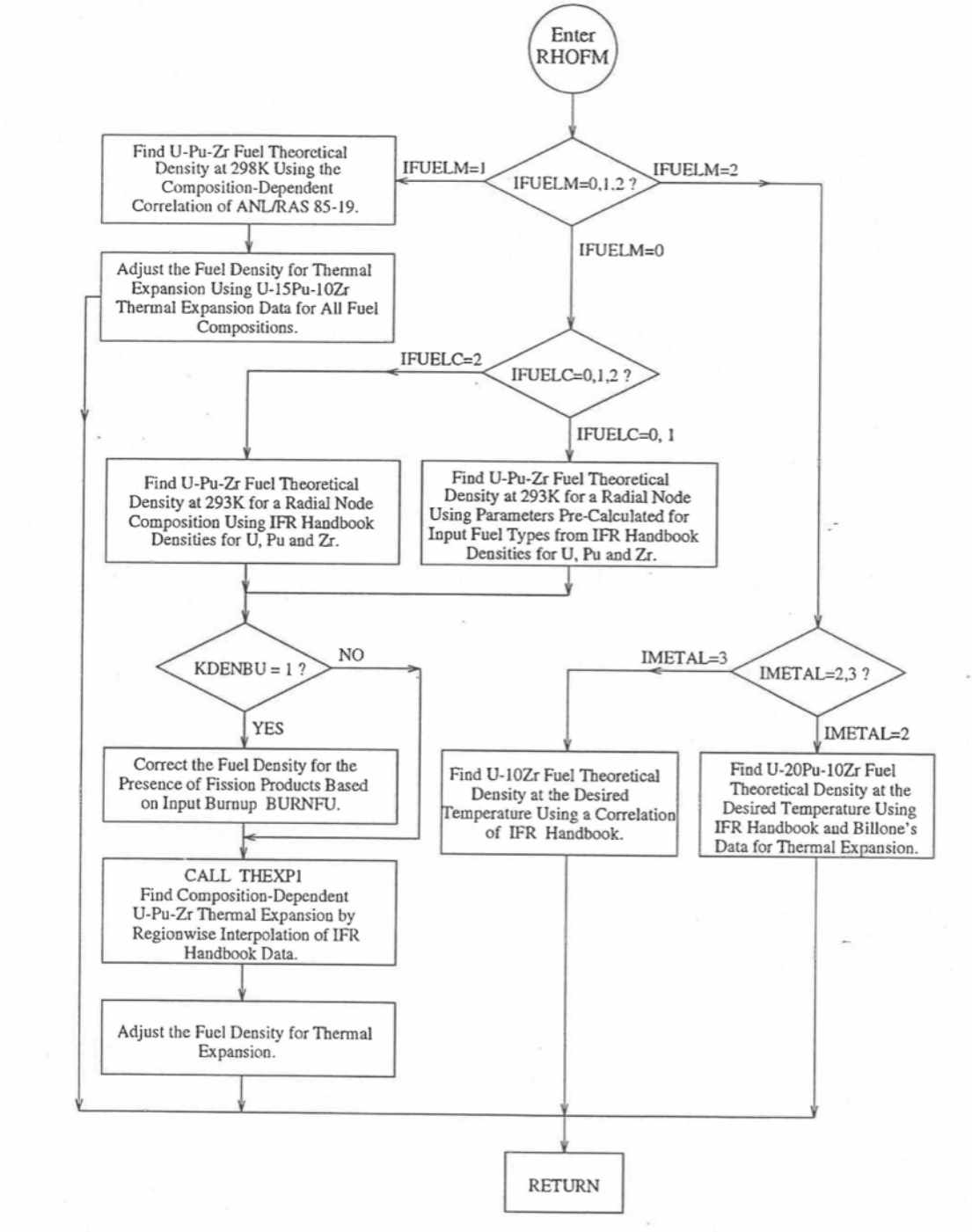


Figure 10.3.37: Flow Diagram of subroutine PHOFM for U-Pu-Zr Fuel Theoretical Density.

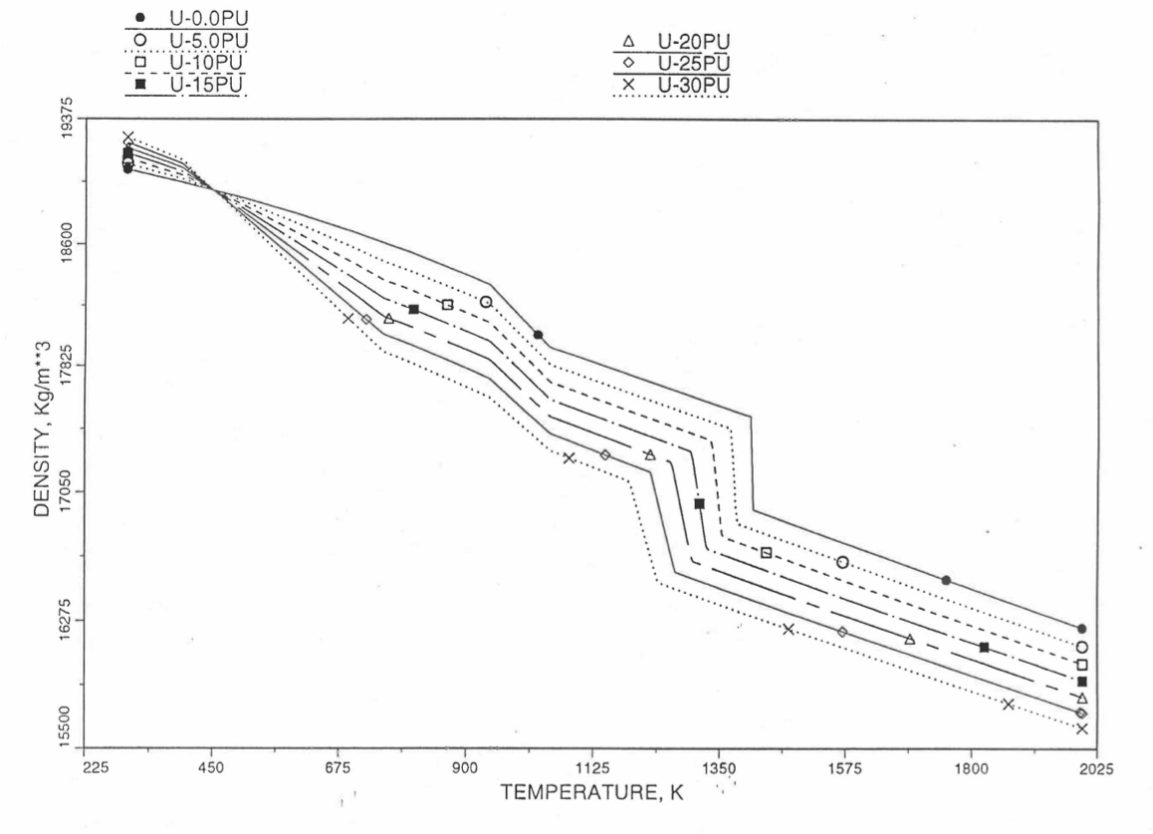


Figure 10.3.38: Theoretical Density of U-Pu Fuel Without Fission Products, Obtained from Regionwise Interpolation.

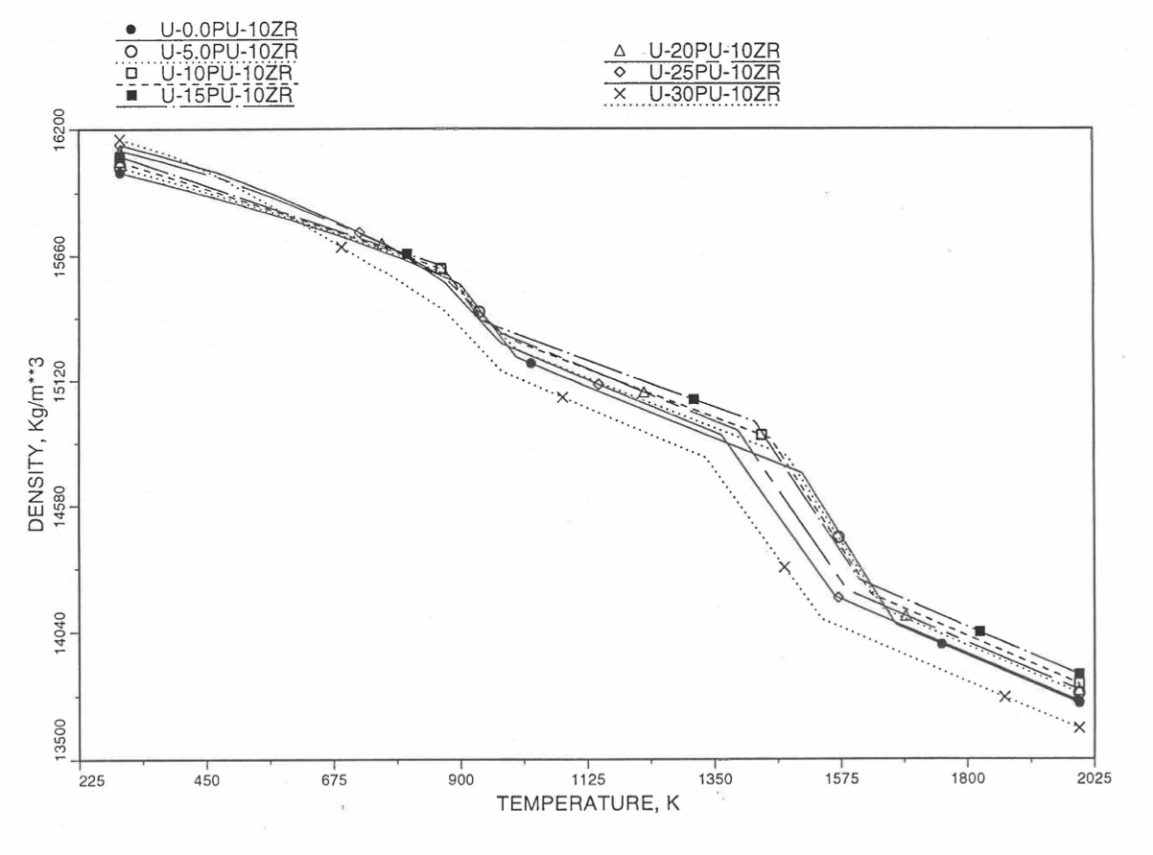


Figure 10.3.39: Theoretical Density of U-Pu-10Zr Fuel Without Fission Products, Obtained From Regionwise Interpolation of IFR Handbook Data.

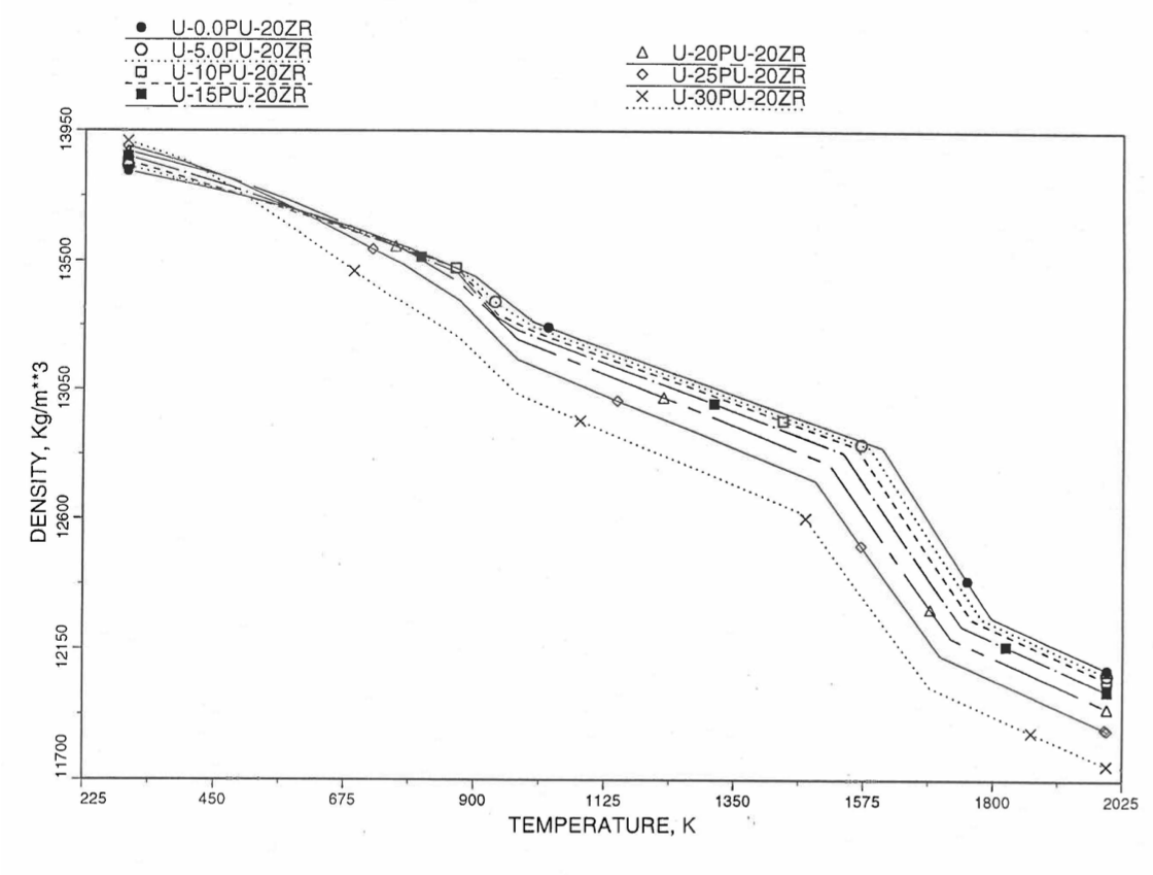


Figure 10.3.40: Theoretical Density of U-Pu-20Zr Fuel Without Fission Products, Obtained from Regionwise Interpolation of IFR Handbook Data.

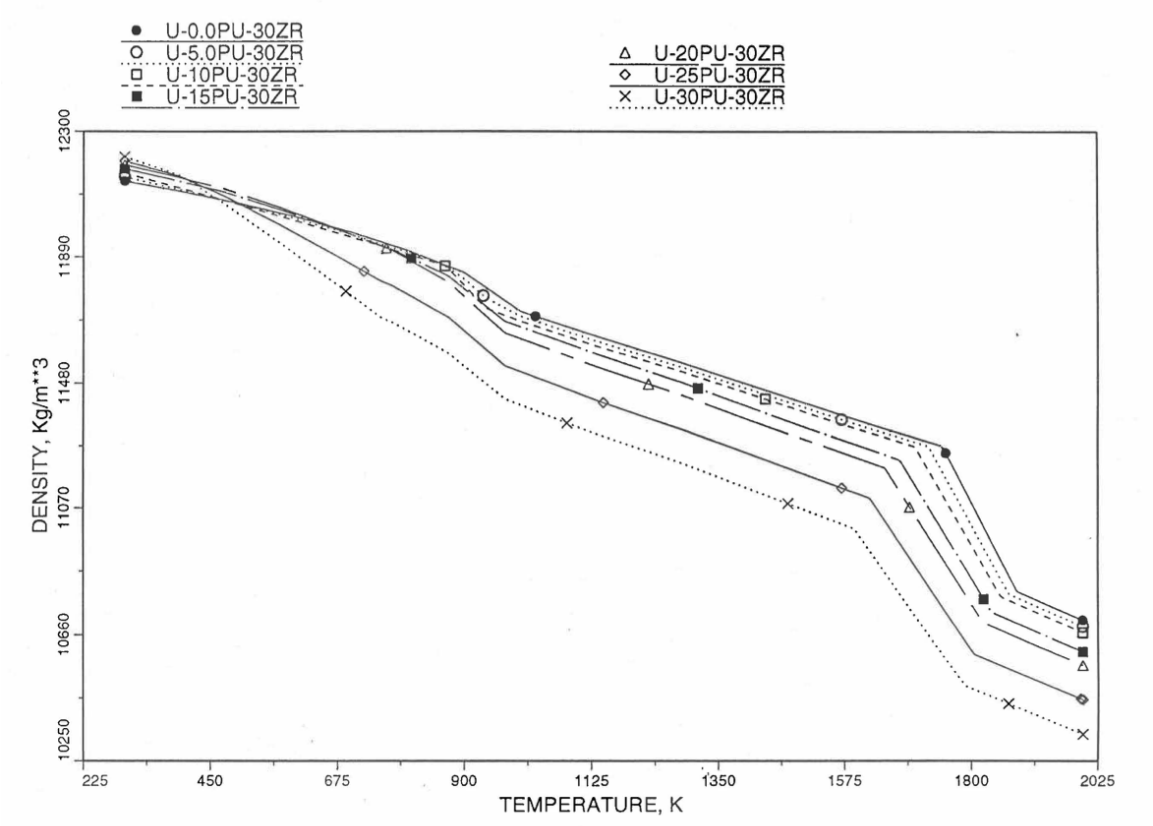


Figure 10.3.41: Theoretical Density of U-Pu-30Zr Fuel Without Fission Products, Obtained from Regionwise interpolation of IFR Handbook Data.

10.3.2.4 Thermal Expansion of U-Pu-Zr Fuel

In this section, a method of interpolating the linear thermal expansion of U-Pu-Zr fuel form 293K to any desired temperature T as a function of fuel composition is described. Only the U, Pu and Zr weight fractions are considered and the effect of fission products on thermal expansion is ignored. This thermal expansion is used in the evaluation of theoretical density of irradiated U-Pu-Zr fuel at any desired temperature, as explained in Section 10.3.2.3.

10.3.2.4.1 Description of the Method

Figure 10.3.42 shows the U-Pu-Zr ternary composition triangle marked with the points representing all the 8 alloys/components for which thermal expansion data re reported in the IFR Handbook [10-8]. The composition points of these 8 database alloys/components are used to divide the whole ternary composition triangle into 9 triangular regions of interpolation. For evaluating the linear thermal expansion of a desired solid U-Pu-Zr alloy represented by a point located in any region of interpolation, the method consists of mixing the 3 database alloys represented by the corners of the region of interpolation. The proportions or weight fractions of the 3 database alloys in mixing are so calculated in the method that the resulting mixture has the composition of the desired alloy. The calculated mixing fractions are then used as weight factors in making a linear combination of (or liner interpolation between) thermal expansions of the 3 database alloys corresponding to the desired temperature, and the value of the linear combination is the interpolated thermal expansion of the desired alloy. It should be noted that the calculated mixing fractions are never negative and the linear combination never requires extrapolation because the desired alloy composition point is always inside the triangle with corners at the 3 database alloy points. If the desired alloy composition point were outside this triangle, the linear combination would involve negative mixing fractions and extrapolation which are avoided entirely in the present method.

10.3.2.4.2 Thermal Expansion Data of Basic Alloys

Table 10.3.3 gives the linear thermal expansion data of the 8 basic solid alloys or components that are used in the present method of evaluating composition-dependent thermal expansion of a U-Pu-Zr alloy. The 8 basic alloys/components are: (1) U metal, (2) U-10Zr, (3) U-20Zr, (4) Zr metal, (5) U-15Pu-10Zr, (6) U-19Pu-10Zr, (7) U-26Pu-10Zr, and (8) Pu metal. The data have all been taken from the Metallic Fuels Handbook [10-8] with the modification that the thermal expansion data of U, Pu and Zr metals between their first and last solid-state transition temperatures are not included in Table 10.3.3, and not used in the present method. The thermal expansion data of U, Pu and Zr metals at their first and last transition temperatures are included in Table 10.3.3, and the thermal expansion is assumed to vary linearly with temperature between the first and last transition temperatures. This is reasonable because the sharp jumps (i.e., discontinuities) in thermal expansion data of U, Pu and Zr metals, over the temperature range of solid-state transitions, are not exhibited by alloys. The value of thermal expansion at 2500K shows in Table 10.3.3, for each database alloy/component has been obtained from the Handbook data for percent thermal expansion and coefficient of thermal expansion at the highest temperature available. A warning message is printed by the SAS4A/SASSYS-1 codes if the thermal expansion value at 2500K of any database alloy is ever used in computation. The uranium thermal expansion data at 1500K, 1600K and 2500K (above its melting point) in Table 10.3.3 have been obtained by subtracting the thermal expansion due to melting (i.e., 1.14%) from the data given in the Handbook. This is done because the data in Table 10.3.3 is used in the present method to interpolate the thermal expansion of a desired U-Pu-Zr alloy up to its solidus temperature only. The evaluation of thermal expansion above the solidus temperature is described below in Section D.

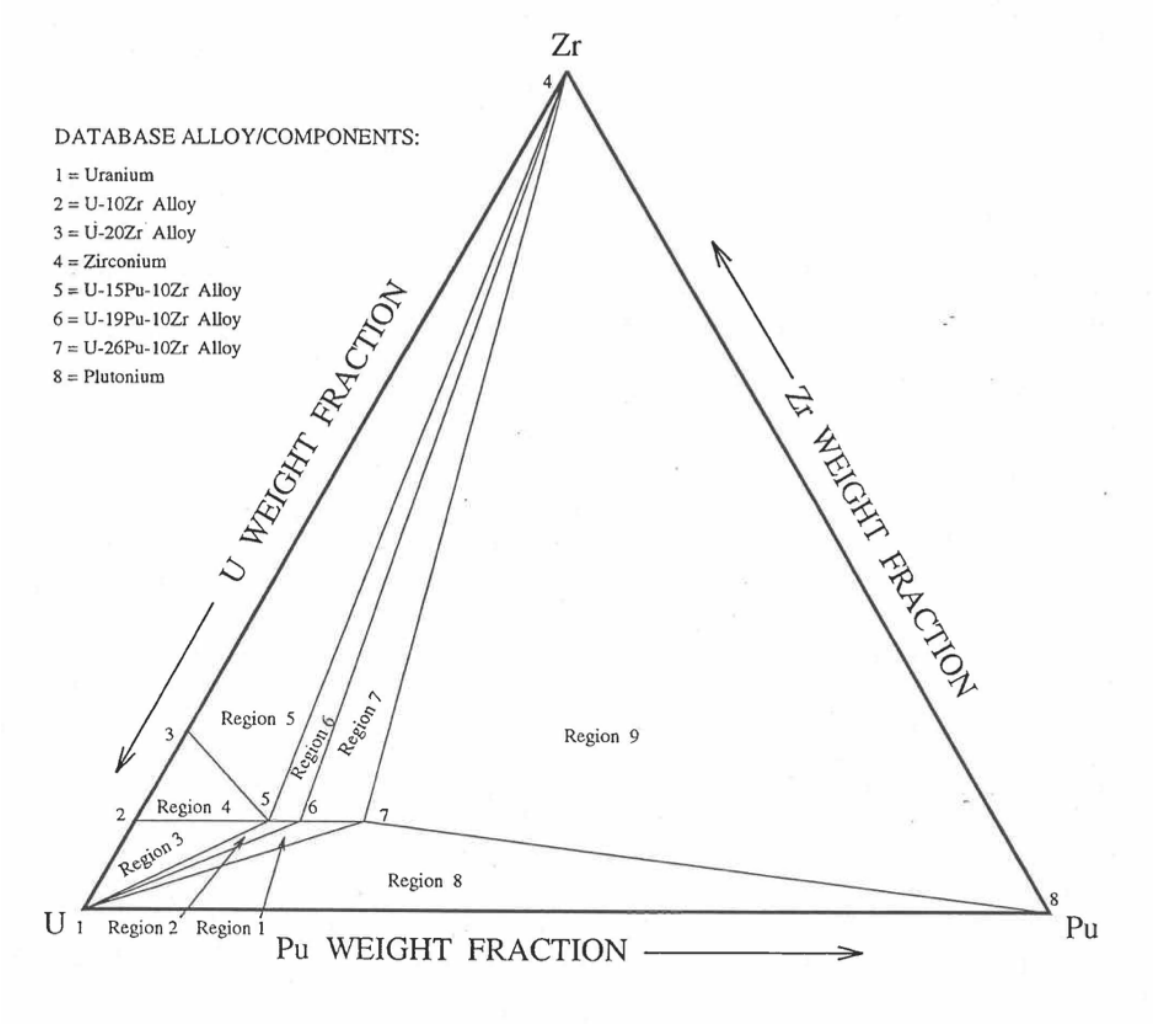


Figure 10.3.42: Diagram Showing Thermal Expansion Interpolation Regions for Solid U-Pu-Zr Alloy Fuels.

10.3.2.4.3 Method of Interpolation

The method of interpolation to evaluate the linear thermal expansion of a U-Pu-Zr fuel composition at and below its solidus temperature consists of first determining the region of the ternary composition triangle (shown in Figure 10.3.42) in which the desired fuel composition lies, and then finding the weight fractions in which the 3 database alloys of the region are required to be mixed to obtain the desired fuel composition. From the mass balance of U, Pu and Zr, the following equations have been obtained in Section 10.3.2.1 for the mixing fractions, X_{i}, of the database alloys that are assumed to be numbered 1, 2 and 3:

Table 10.3.3: Linear Thermal Expansion Data of the 8 Basic Solid Alloys or Components Used in the Method of Regionwise Interpolation

No.	1 Temp (K)	Uranium $\Delta L/L_0$ (%)	2 Temp (K)	U-10Zr $\Delta L/L_0$ (%)	3 Temp (K)	U-20Zr $\Delta L/L_0$ (%)	4 Temp (K)	Zirconium $\Delta L/L_0$ (%)
1	293	0	293	0	293	0	293	0
2	400	0.157	400	0.142	400	0.107	400	0.06
3	500	0.315	500	0.281	500	0.219	500	0.123
4	600	0.494	600	0.433	600	0.348	600	0.192
5	700	0.697	700	0.603	700	0.5	700	0.265
6	800	0.924	800	0.799	800	0.681	800	0.343
7	900	1.186	900	1.027	900	0.899	900	0.442
8	941	1.3	1000	1.725	1000	1.3	1000	0.505
9	1048	2.05	1100	1.95	1100	1.487	1100	0.586
10	1100	2.168	1200	2.175	1200	1.674	1137	0.617
11	1200	2.398	2500	5.1	2500	4.105	1284	0.617
12	1400	2.855					1400	0.725
13	1408	2.866					1600	0.922
14	1500	3.092					1800	1.138
15	1600	3.362					2500	1.803
16	2500	5.657						

No.	5 Temp (K)	U-15Pu-10Zr $\Delta L/L_0$ (%)	6 Temp (K)	U-19Pu-10Zr $\Delta L/L_0$ (%)	7 Temp (K)	U-26Pu-10Zr $\Delta L/L_0$ (%)	8 Temp (K)	Plutonium $\Delta L/L_0$ (%)
1	293	0	293	0	293	0	293	0
2	400	0.18	373	0.103	373	0.108	350	0.279
3	500	0.356	473	0.242	473	0.253	395	0.502
4	600	0.532	573	0.431	573	0.455	753	6.287
5	700	0.708	673	0.638	673	0.683	800	6.459
6	800	0.884	773	0.86	773	0.924	2500	12.68
7	868	1.004	873	1.146	873	1.276		
8	938	1.524	973	1.662	973	1.868		
9	1000	1.649	1073	1.881	1073	2.095		
10	1100	1.85	1173	2.104	1173	2.327		
11	1200	2.051	1273	2.334	1233	2.464		
12	1300	2.252	2500	5.113	2500	5.386		
13	1378	2.409						
14	2500	4.664						
15								
16								

$$X_{\mathrm{i}} = \frac{D_{\mathrm{i}}}{D}, \quad i = 1, 2, 3 \tag{10.3-45}$$

$$D = (W_{\mathrm{z1}} - W_{\mathrm{z2}})(W_{\mathrm{p3}} - W_{\mathrm{p1}}) - (W_{\mathrm{p1}} - W_{\mathrm{p2}})(W_{\mathrm{z3}} - W_{\mathrm{z1}}) \tag{10.3-46}$$

$$D_1 = (W_{\mathrm{z}} - W_{\mathrm{z2}})(W_{\mathrm{p3}} - W_{\mathrm{p}}) - (W_{\mathrm{p}} - W_{\mathrm{p2}})(W_{\mathrm{z3}} - W_{\mathrm{z}}) \tag{10.3-47}$$

$$D_2 = (W_{\mathrm{z1}} - W_{\mathrm{z}})(W_{\mathrm{p3}} - W_{\mathrm{p1}}) - (W_{\mathrm{p1}} - W_{\mathrm{p}})(W_{\mathrm{z3}} - W_{\mathrm{z1}}) \tag{10.3-48}$$

$$D_3 = (W_{\mathrm{z1}} - W_{\mathrm{z2}})(W_{\mathrm{p}} - W_{\mathrm{p1}}) - (W_{\mathrm{p1}} - W_{\mathrm{p2}})(W_{\mathrm{z}} - W_{\mathrm{z1}}) \tag{10.3-49}$$

where

$W_{\mathrm{ui}}, W_{\mathrm{pi}}, W_{\mathrm{zi}}$ = weight fractions of U, Pu and Zr in database alloy I of the region in which the desired fuel compositions lies,

$W_{\mathrm{u}}, W_{\mathrm{p}}, W_{\mathrm{z}}$ = weight fractions of U, Pu and Zr in the desired fuel composition.

The sum of X_1, X_2 and X_3 is always 1. In case of interpolation, the values of X_1, X_2 and X_3 are always positive. If any extrapolation were involved, the value of at least one weight fraction would be negative which is entirely avoided in the present method. The linear thermal expansion of the desired U-Pu-Zr fuel composition from 293K to a temperature T at or below its solidus is then evaluated by mass-weighted averaging of the thermal expansions of the 3 database alloys from 293K to the same temperature T.

$$\frac{\Delta L\,(T, W_{\mathrm{p}}, W_{\mathrm{z}})}{L_{\mathrm{o}}} = X_1 \frac{\Delta L_1}{L_{\mathrm{o}}} + X_2 \frac{\Delta L_2}{L_{\mathrm{o}}} + X_3 \frac{\Delta L_3}{L_{\mathrm{o}}} \tag{10.3-50}$$

where

$\frac{\Delta L_{\mathrm{i}}}{L_{\mathrm{o}}}$ = linear thermal expansion from 293K to temperature T of the database alloy i of the region in which the desired fuel composition lines.

10.3.2.4.4 Thermal Expansion During and After Melting

Thermal expansion data during and after melting is available for none of the database alloys except uranium metal. Therefore, the thermal expansion of all alloy compositions above their solidus temperatures is evaluated using the uranium metal data, and a warning message is printed by the SAS4A/SASSYS-1 codes if this is ever required in computation. The linear thermal expansion of uranium during melting at 1408K reported in the Handbook [10-8] is 1.14%, and the coefficient of linear expansion of molten uranium is 25.5 x 10-6/K. In the present method, the post-solidus uranium thermal expansion is added to Eq. (10.3-50) to obtain the following equation for estimating the linear thermal expansion form 293K to a temperature T above the solidus of the desired alloy:

$$\begin{aligned}\frac{\Delta L\left(T, W_{\mathrm{p}}, W_{\mathrm{z}}\right)}{L_{\mathrm{o}}} &= X_1 \frac{\Delta L\left(T_{\mathrm{s}}, W_{\mathrm{p}}, W_{\mathrm{z}}\right)}{L_{\mathrm{o}}} + X_2 \frac{\Delta L_2\left(T_{\mathrm{s}}, W_{\mathrm{p}}, W_{\mathrm{z}}\right)}{L_{\mathrm{o}}} \\ +X_3 \frac{\Delta L_3\left(T_{\mathrm{s}}, W_{\mathrm{p}}, W_{\mathrm{z}}\right)}{L_{\mathrm{o}}} &+ 0.0114 \min\left(\frac{T - T_{\mathrm{s}}}{T_{\mathrm{l}} - T_{\mathrm{s}}}, 1\right) + 25.5 \times 10^{-6}\left(T - T_{\mathrm{s}}\right)\end{aligned} \tag{10.3-51}$$

where

$T_{\mathrm{s}}, T_{\mathrm{l}}$ = solidus and liquidus temperatures of the desired alloy.

The fourth term on the right hand side is basically the product of alloy melt fraction and uranium thermal expansion during melting phase change, and the fifth term is an estimate of the molten alloy thermal expansion due to temperature rise above the alloy solidus.

10.3.2.4.5 Subroutine THEXP1 and its Comparison With the Handbook Data

Figure 10.3.43 shows flow diagram of the subroutine THEXP1 developed for calculating the linear thermal expansion, based on this method of regionwise interpolation, of U-Pu-Zr fuel as a function of composition. This subroutine is used, in the SAS4A/SASSYS-1 codes, in the calculation of the theoretical density as a function of temperature and composition, given by Eq. (10.3-44).

Figure 10.3.44 to Figure 10.3.51 show the comparison of calculated linear thermal expansion with all the data in the Metallic Fuels Handbook [10-8] for the 8 basic alloys or components discussed in Section B. It can be noted in Figure 10.3.44, Figure 10.3.47 and Figure 10.3.51 that the Handbook data for U, Zr and Pu metals over the solid-state transition temperature range differ from the calculated values. As discussed in Section B, the calculated thermal expansion varies linearly with temperature between the first and last solid state transitions. In the case of Zr and Pu (Figure 10.3.47 and Figure 10.3.51), this linearization has led to ignoring the thermal contractions (taking place during heatup) that occur at solid state transitions. However the behavior of the 3 metals when alloyed is different from their behavior in isolation, and the linearization is the expected behavior when alloyed.

Figure 10.3.52 to Figure 10.3.54 show the variation of linear thermal expansion with temperature for 21 alloy compositions obtained from this subroutine. The Pu and Zr contents each vary from zero to 30 wt % in the alloy compositions plotted in the figures. The solidus and liquidus temperatures required as input to the THEXP1 subroutine were calculated using another subroutine developed by Pelton [10-13]. The linear thermal expansion from 293K to solidus varies over the range 2.4% to 4.0% for the alloy compositions plotted.

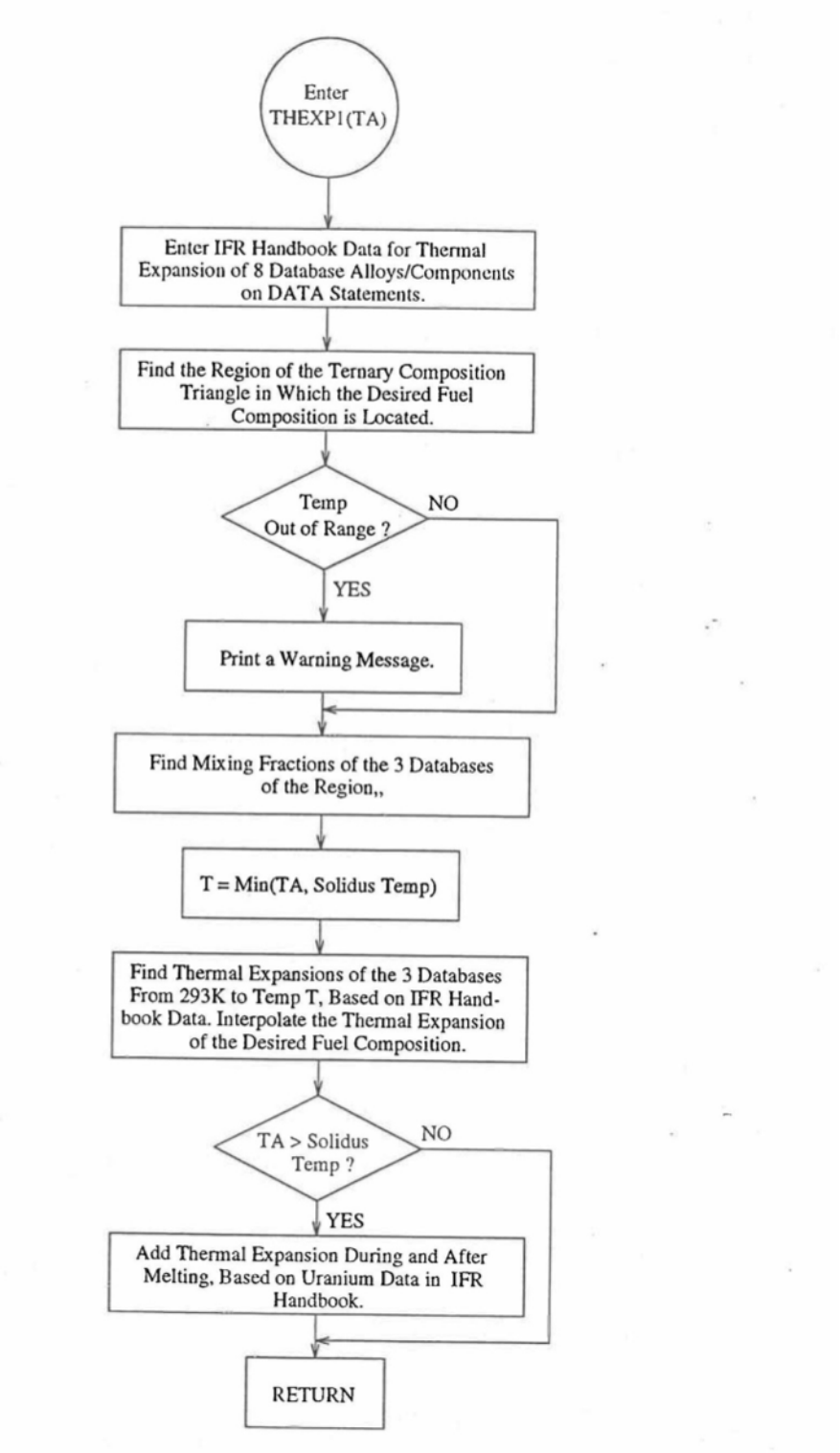


Figure 10.3.43: Flow Diagram of Subroutine THEXP2 for U-Pu-Zr Thermal Expansion.

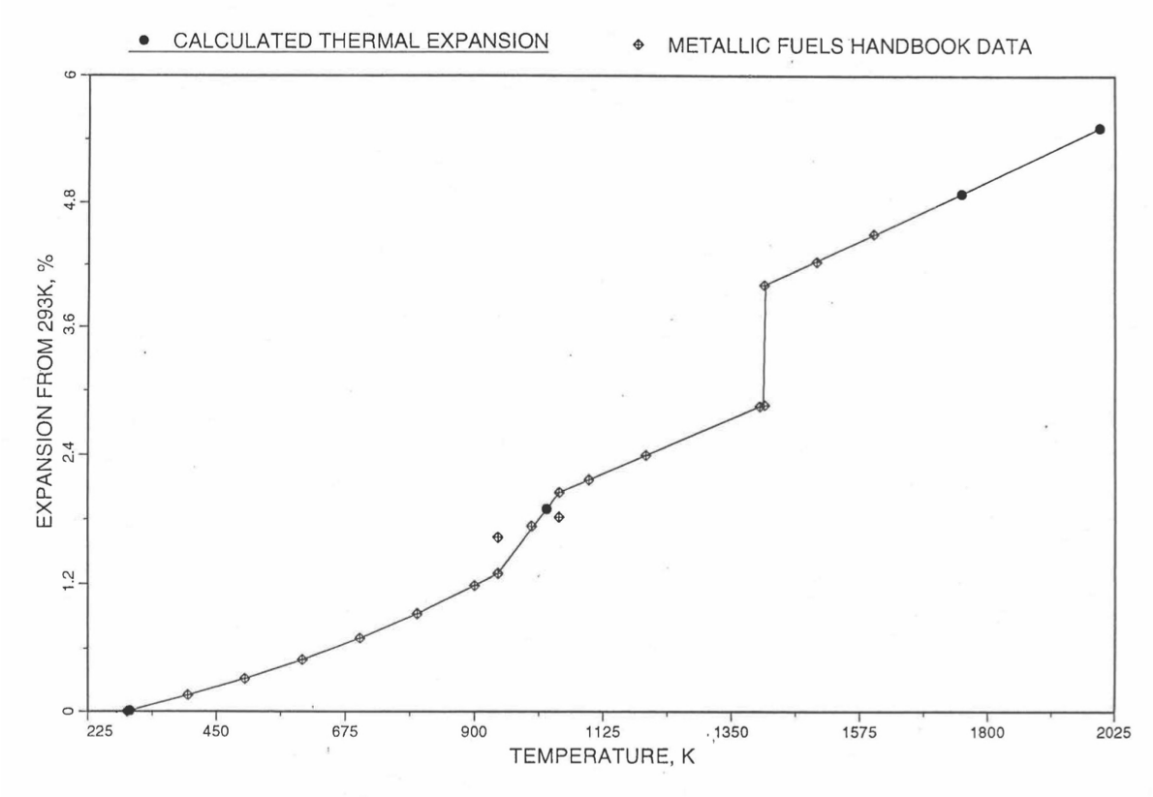


Figure 10.3.44: Comparison of Uranium Metal Linear Thermal Expansion Data in IFR Handbook With Those Calculated by the Regionwise Interpolation Subroutine THEXP1.

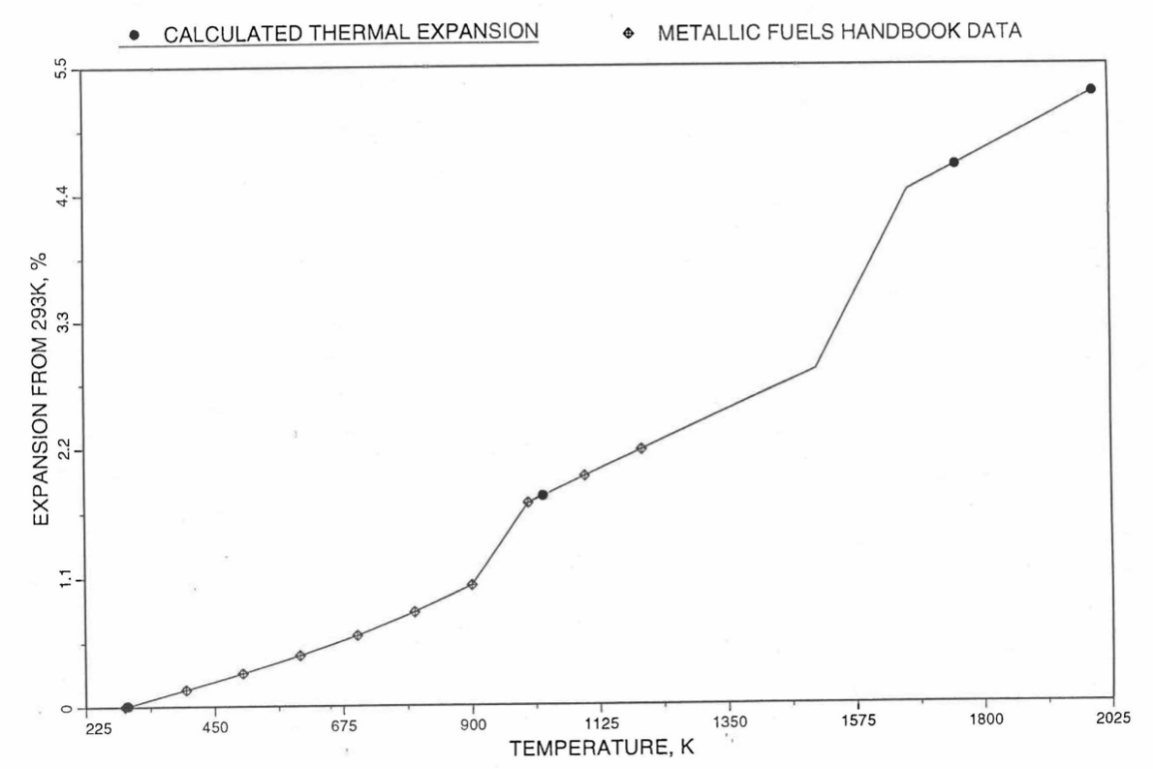


Figure 10.3.45: Comparison of U-10Zr Alloy Linear Thermal Expansion Data in IFR Handbook With Those Calculated by the Regionwise Interpolation Subroutine THEXP1.

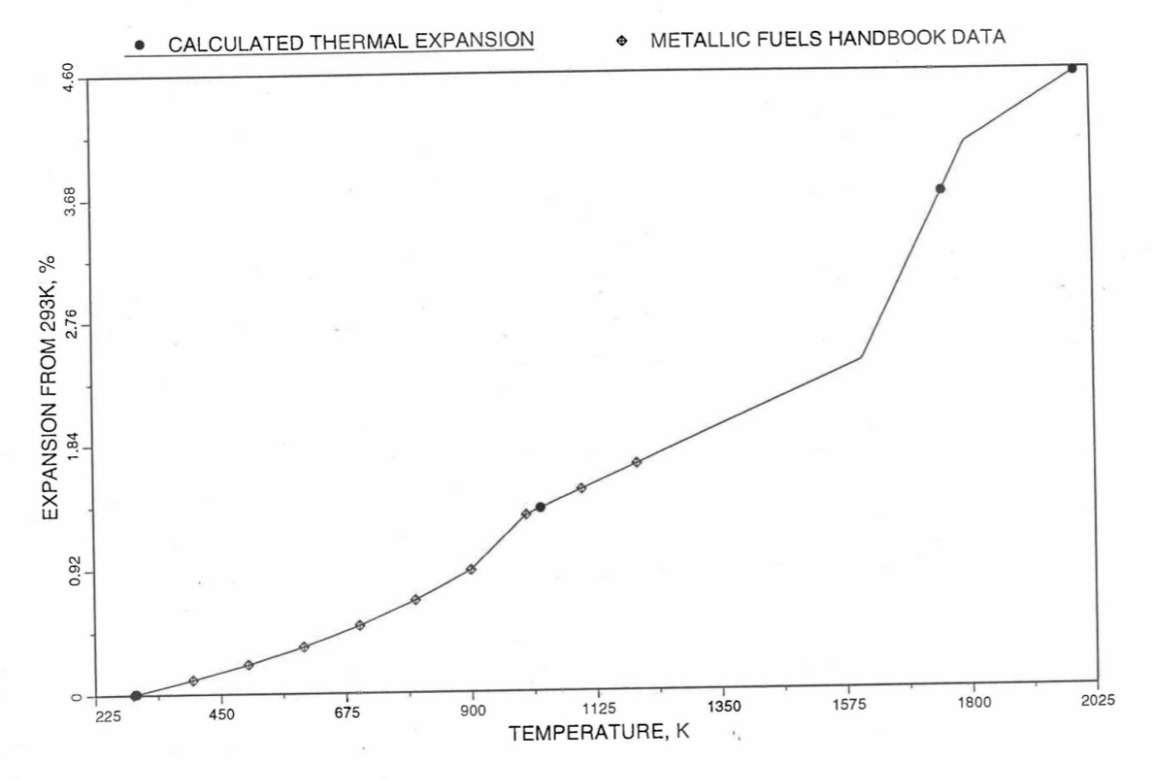


Figure 10.3.46: Comparison of U-20Zr Alloy Linear Thermal Expansion Data in IFR Handbook With those Calculated by the Regionwise Interpolation Subroutine THEXP1.

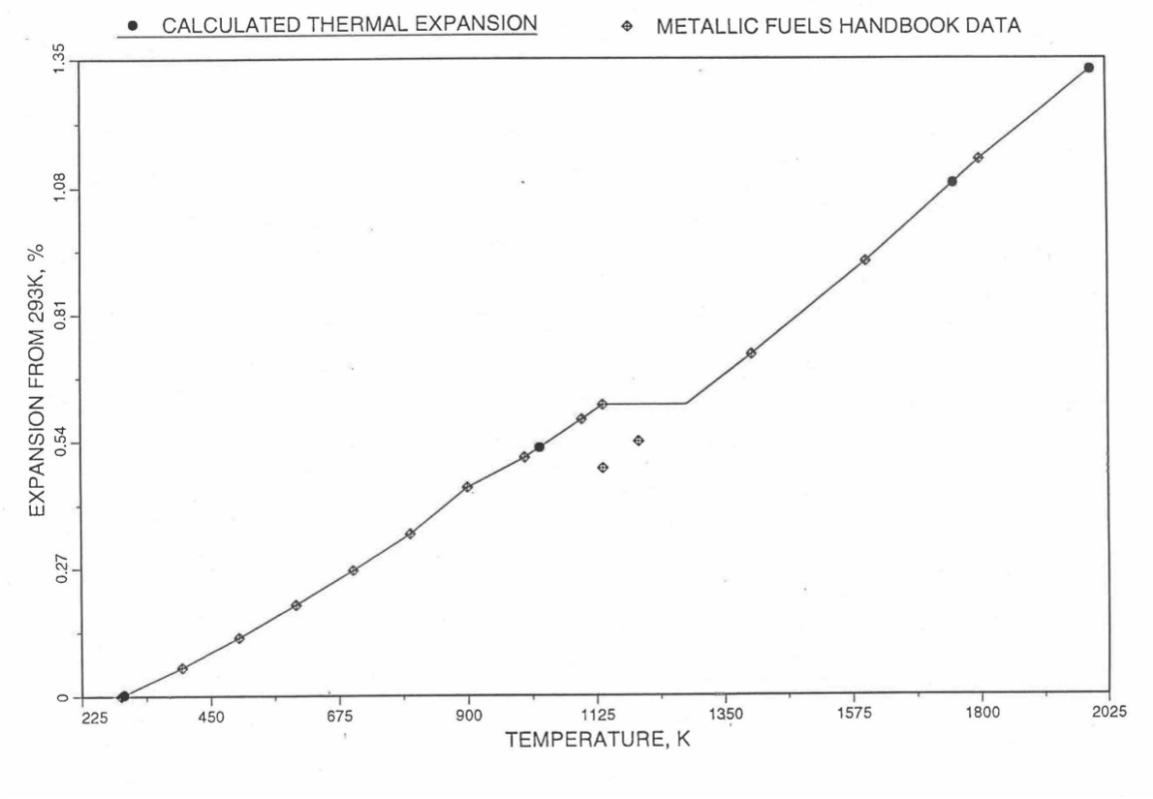


Figure 10.3.47: Comparison of Zirconium Metal Linear Thermal Expansion Data in IFR Handbook With Those Calculated by the Regionwise Interpolation Subrountine THEXP1.

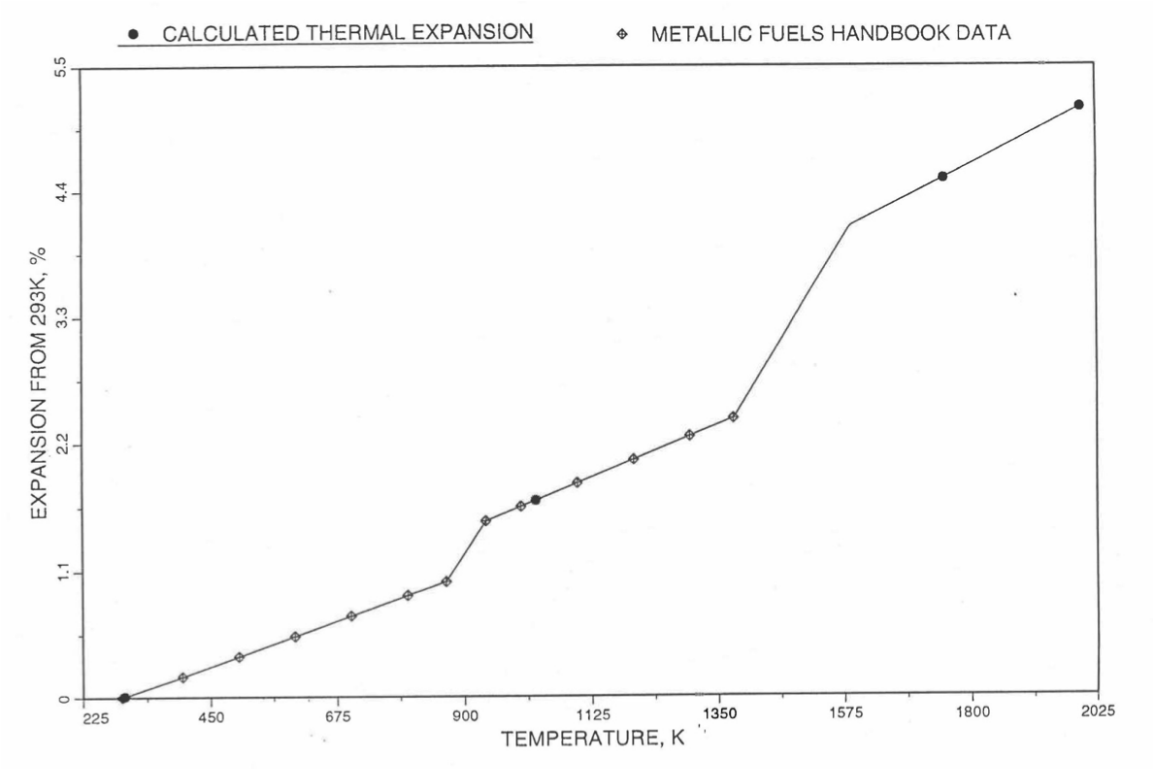


Figure 10.3.48: Comparison of U-15Pu-10Zr Linear Thermal Expansion Data in IFR Handbook With Those Calculated by the Regionwise Interpolation Subroutine THEXP1.

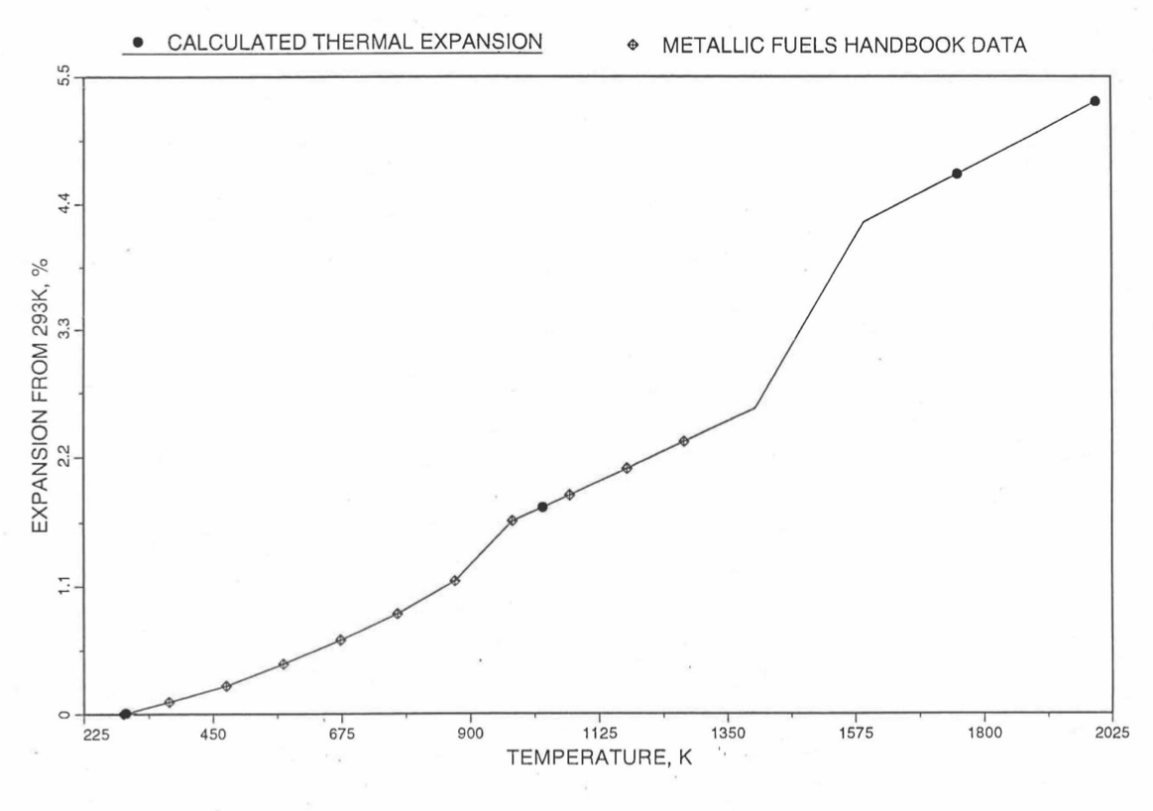


Figure 10.3.49: Comparison of U-19Pu-10Zr Linear Thermal Expansion Data in IFR Handbook With Those Calculated by the Regionwise Interpolation Subroutine THEXP1.

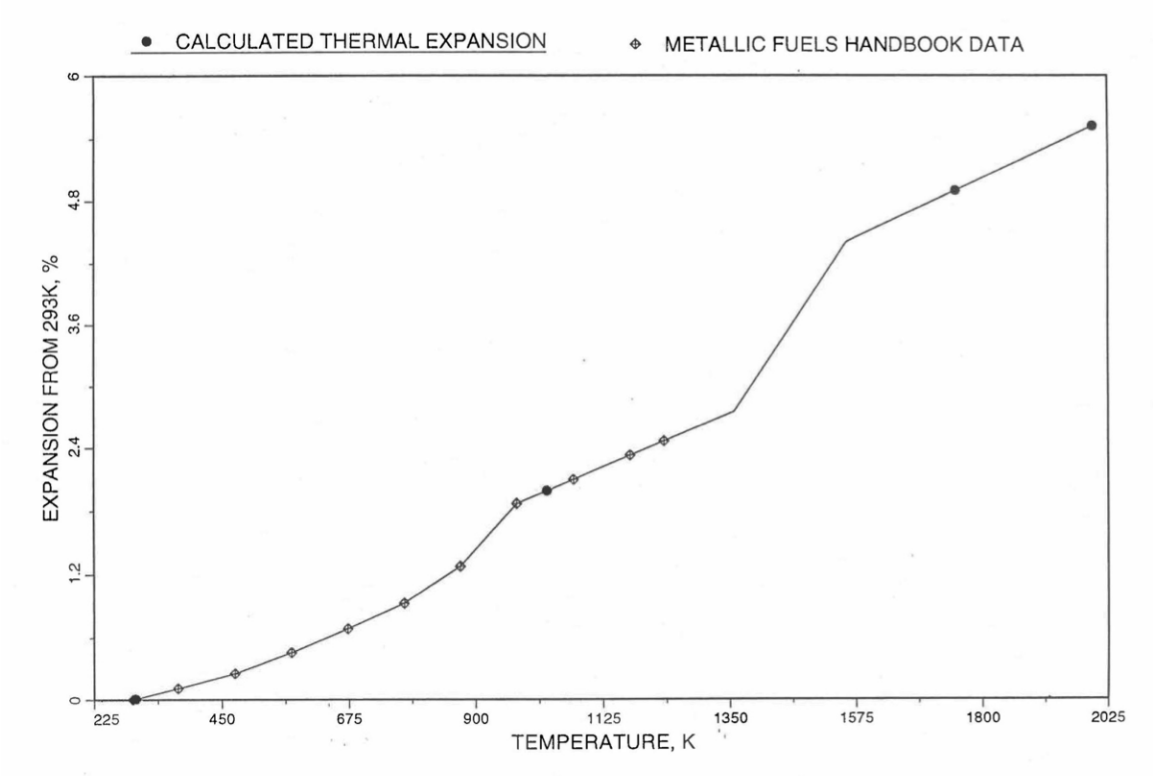


Figure 10.3.50: Comparison of U-26Pu-10Zr Linear Thermal expansion Data in IFR Handbook With those Calculated by the Regionwise Interpolation Subroutine THEXP1.

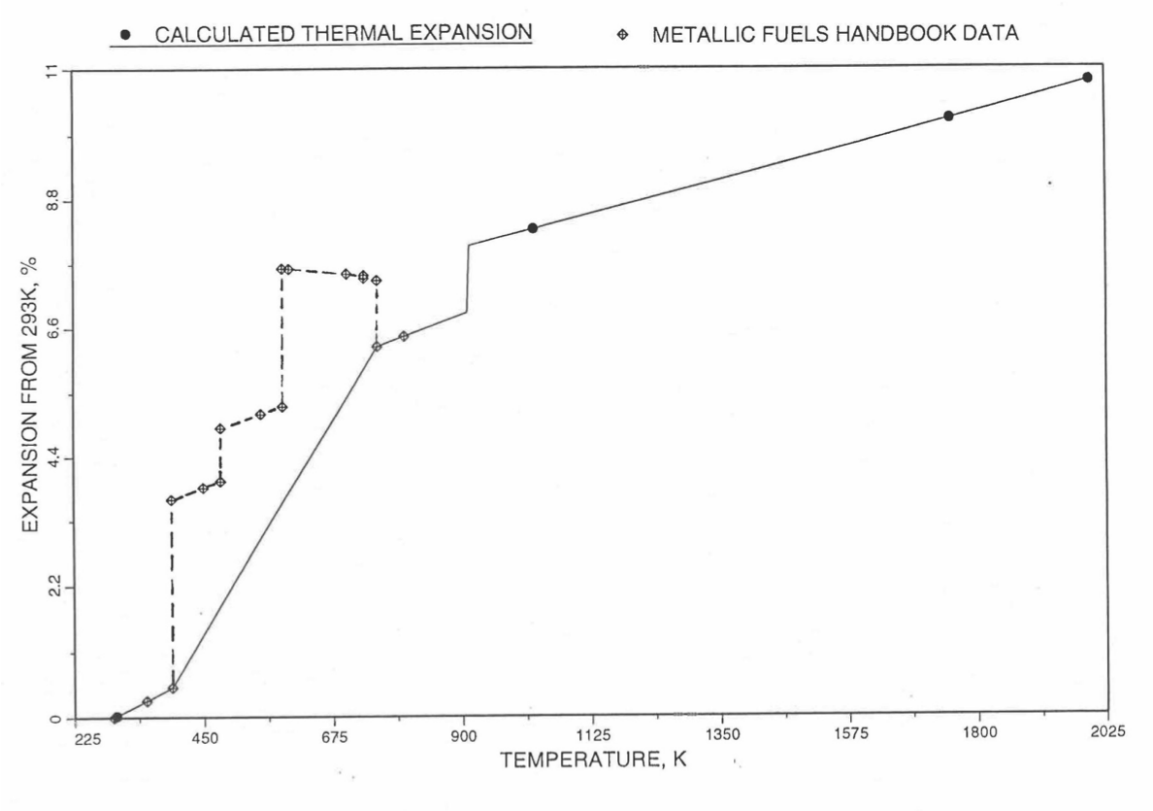


Figure 10.3.51: Comparison of Plutonium Metal Liner Thermal Expansion Data in IFR Handbook With Those calculated by the Regionwise Interpolation Subroutine THEXP1.

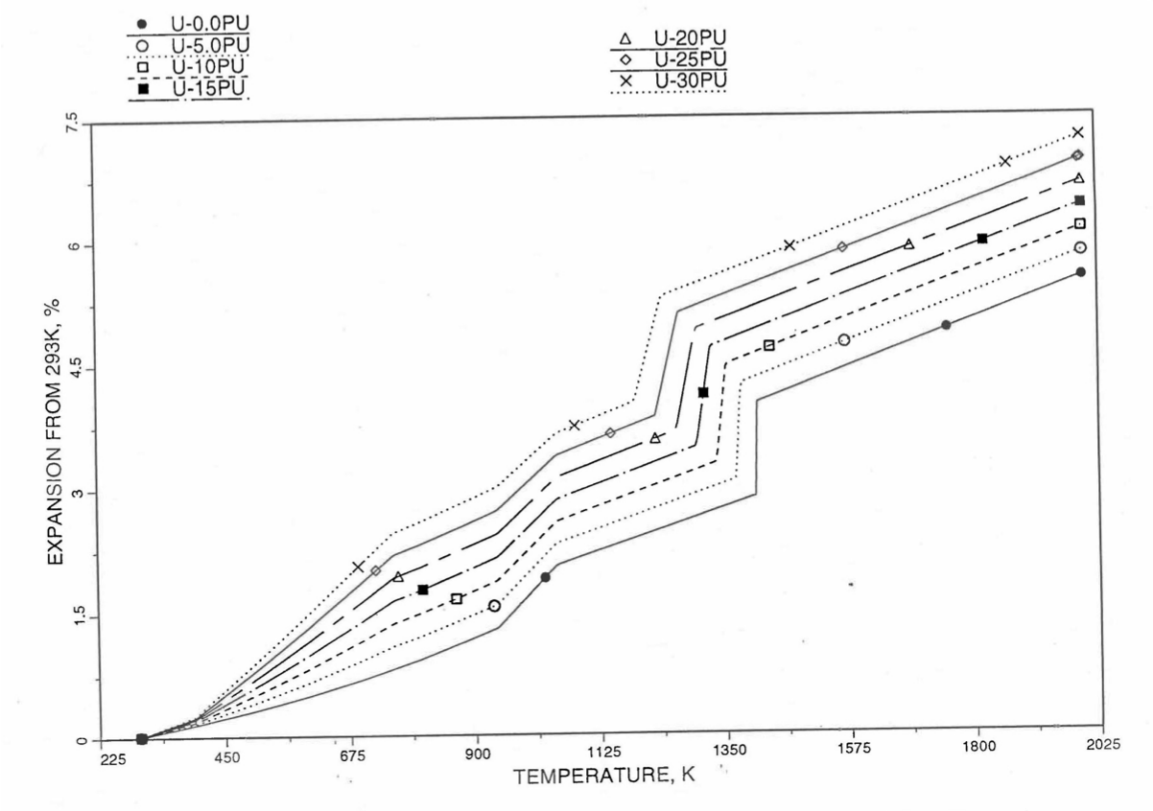


Figure 10.3.52: Linear Thermal Expansion of U-Pu Fuel Without Fission Products, Obtained From Regionwise Interpolation of IFR Handbook Data.

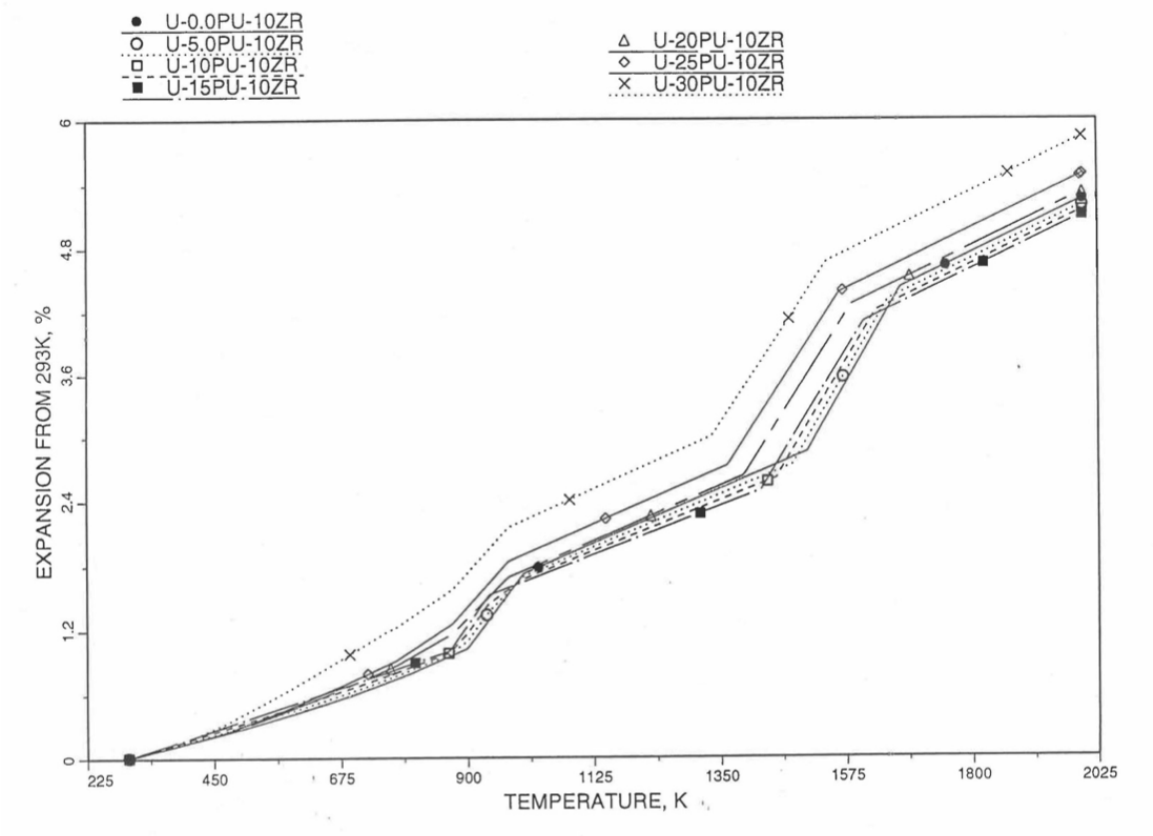


Figure 10.3.53: Linear Thermal Expansion of U-Pu-10Zr fuel Without Fission Products, Obtained From Regionwise Interpolation of IFR Handbook Data.

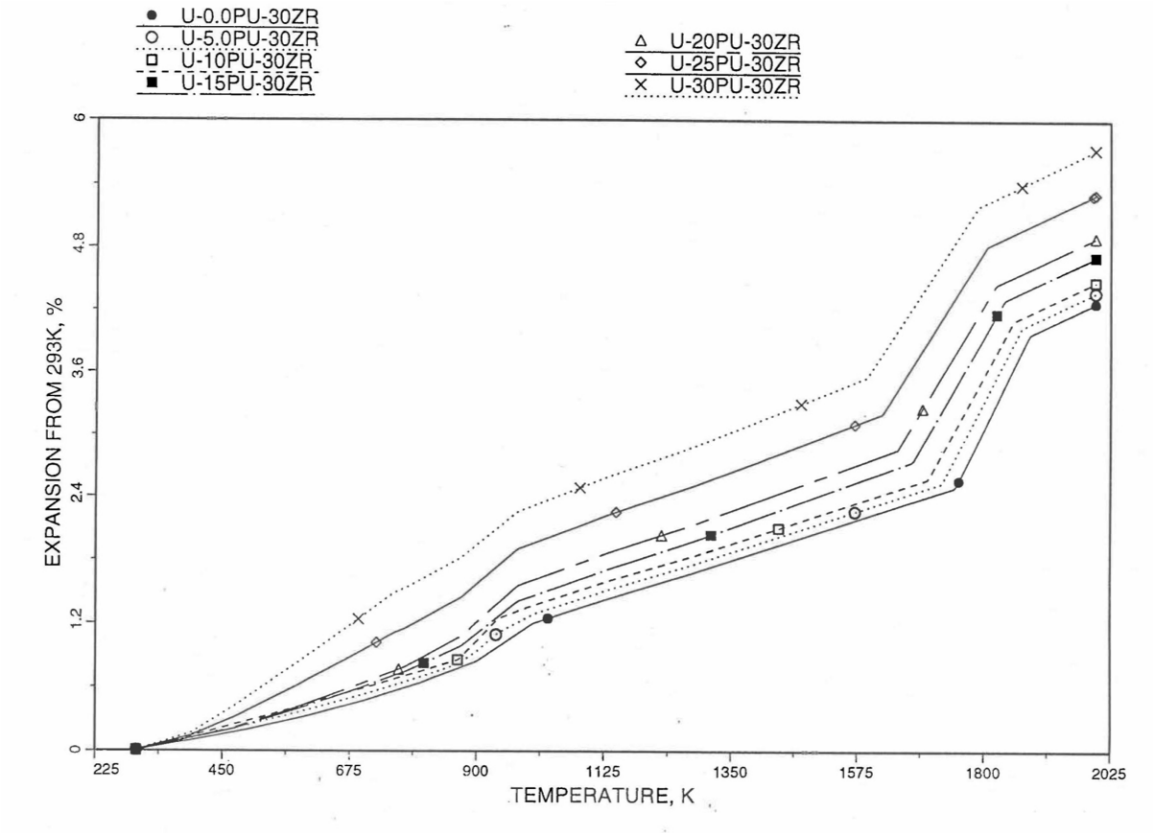


Figure 10.3.54: Linear Thermal Expansion of U-Pu-30Zr Fuel Without Fission Products, Obtained From Regionwise Interpolation of IFR Handbook Data.

10.3.2.5 Thermal Conductivity of Unirradiated 100% Dense U-Pu-Zr Fuel

10.3.2.5.1 Introduction

To perform calculations for the Mark-V fuel (U-20Pu-10Zr), and other metallic alloy fuels to different compositions, especially the fuel compositions across the pin radius resulting from the migration of alloy constituents during steady state irradiation, there is a need for modeling and correlation development for the thermal conductivity of U-Pu-Zr fuels as a function of composition, using the data presented in the IFR Metallic Fuels Handbook [10-8, 10-9].

Earlier, before the preparation of the Metallic Fuels Handbook [10-9], a correlation was developed to estimate the composition- and temperature-dependent thermal conductivity of 100% dense solid U-Pu-Zr alloy fuel [10-4]. In this correlation, the thermal conductivity of the U-Zr binary alloy having the uranium-to-zirconium atomic ratio of the desired U-Pu-Zr fuel was first evaluated from the data for U metal and U-50 at. % Zr alloy. This thermal conductivity was then adjusted for the presence of Pu by subtracting a term linearly proportional to Pu atom fraction where the coefficient of proportionality was determined (as a function of temperature and zirconium atom fraction) by fitting to the thermal conductivity data available for three U-Pu-Zr alloys, i.e., U-16.2Pu-6.2Zr, U-14.7Pu-9.7Zr and U-18.4Pu-11.5Zr. This correlation is valid when Pu atom fraction is less than 0.2, and Zr atom fraction is less than 0.5. The variation of thermal conductivity obtained from this correlation has been plotted for alloy compositions having 0.0 to 30 wt % Pu and 1.5 to 40 wt % Zr [10-17]. A drawback of this correlation [10-4] is that, for some alloy compositions, the thermal conductivity becomes negative (unphysical) at room temperature, and shows a maximum when plotted as a function of temperature (an unexpected behavior [10-18] for temperatures greater than 300 K). Although the allow compositions showing this unphysical/unexpected behavior are out of the noted region of validity, the correlation has another drawback that the thermal conductivity data used in obtaining the correlation were taken from sources [10-14, 10-19, 10-20] other than the Metallic Fuels Handbook.

The Metallic Fuels Handbook [10-8, 10-9] presents thermal conductivity data for U metal and a number of U-Zr and U-Pu-Zr alloys. It also gives a correlation to estimate the composition- and temperature-dependent thermal conductivity of 100% dense solid U-Pu-Zr alloys. In addition, Billone has developed, based on the data reported in the IFR Handbook [10-8], two other correlations [10-21, 10-22] for use in the analysis of EBR-II Mark-V fuel core, and so the following 3 correlations based on the Handbook data are available to estimate 100% dense solid U-Pu-Zr fuel thermal conductivity:

1. The correlation in the IFR Handbook [10-8, 10-9],
2. The correlation in Billone's memorandum [10-21] of March 8, 1991,
3. The correlation in Billone's memorandum [10-22] of October 21, 1991.

The above 3 correlations are summarized here in Section 10.7.2 for easy reference. The variation of thermal conductivity obtained form these 3 correlations have been plotted for alloy compositions having 0.0 to 30 wt % Pu and 1.5 to 40 wt % Zr [10-17]. The correlation in March 8 memorandum should not be used because it is not well-behaved, and so recommended by Billone [10-22]. The thermal conductivity obtained from this correlation, with increasing Pu content at a fixed Zr content, first increases above the U-Zr binary value, then starts decreasing and goes below the binary value, and finally starts increasing again and becomes much greater than the binary value. A drawback of the other two correlations (i.e., the correlations in the IFR Handbook and in the October 21 memorandum) is that the thermal conductivities become negative (unphysical) at room temperature for Pu and Zr contents greater than 20 wt %. Another drawback of the correlation in October 21 memorandum is that the thermal conductivity shows a maximum when plotted as a function of temperature (an unexpected behavior [10-18] for temperatures greater than 300 K). A comparison of these 3 correlations with the Handbook data for U-Zr binary alloys shows that the 3 correlations are almost identical for binary alloys [10-17], and that they agree with the data closely. The correlation in the IFR Handbook can be reliably used for binary alloys containing 1.5 to 20 wt. % Zr.

Recently, the method of estimating U-Pu-Zr alloy enthalpy by regionwise interpolation has been extended to estimate thermal conductivity of 100% dense solid U-Pu-Zr alloy fuels as a function of composition. Based on this method of calculating composition- and temperature-dependent thermal conductivity, a subroutine KFUEL1 has been developed that removes all the drawbacks mentioned above. The subroutine makes use of the thermal conductivity data of 9 basic alloys/components, compressing of all the data in the IFR Handbook [10-8, 10-9]. The U-10Pu data used in the method are obtained from Billone [10-21], and the Pu data are reported in the Plutonium Handbook [10-23]. The 9 basic alloy compositions are used to divide the ternary composition triangle into 10 regions of interpolation as shown in Figure 10.3.55. The method is flexible enough to make use of the thermal conductivity data of additional alloys (as they become available) to divide the composition triangle into a larger number of smaller regions, thus improving its accuracy. The subroutine KFUEL1 has been incorporated into the SAS4A/SASSYS-1 codes.

The technique used at present to estimate the fuel thermal conductivity during and after melting is also discussed in this section. Since all the thermal conductivity data reported in the IFR Handbook lies in the 293 K to 1173 K temperature range, a warning message is printed by the codes whenever fuel thermal conductivity is calculated at a temperature smaller than 293 K or greater than 1200 K.

10.3.2.5.2 Description of the Method

Figure 10.3.55 shows the U-Pu-Zr ternary composition triangle marked with the points representing 9 basic alloys/components comprising of all the thermal conductivity data reported in the IFR Handbook [10-8, 10-9]. The U-10Pu data used here are obtained from Billone [10-21], and the Pu data are reported in the Plutonium Handbook [10-23], and a warning message is printed by the codes when these two data are actually used in any calculation. The composition points of these 9 database alloys/components are used to divide the whole ternary composition triangle into 10 triangular regions of interpolation as shown in Figure 10.3.55. For evaluating the thermal conductivity of a desired solid U-Pu-Zr alloy represented by a point located in any region of interpolation, the method consists of mixing the 3 database alloys represented by the corners of the region of interpolation. The proportions or weight fractions of the 3 database alloys in mixing are so calculated in the method that the resulting mixture has the composition of the desired alloy. The calculated mixing fractions are then used as weight factors in making a linear combination of (or linear interpolation between) thermal conductivity data of the 3 database alloys, and the linear combination gives the interpolated thermal conductivity of the desired alloy. The thermal conductivity data of each basic alloy/component are least-square fitted by aquadratic function of temperature.

$$K_{0\mathrm{i}} = A_{\mathrm{i}1} + A_{\mathrm{i}2}T + A_{\mathrm{i}3}T^2 \tag{10.3-52}$$

where

$K_{0\mathrm{i}}$ = thermal conductivity of the ith 100%-dense solid database alloy/component, W/m-K,

$A_{\mathrm{i}1}, A_{\mathrm{i}2}, A_{\mathrm{i}3}$ = coefficients of the quadratic for the thermal conductivity of the ith database alloy,

T = alloy temperature, K.

The interpolation of the thermal conductivity data of the 3 basic alloys is performed by interpolating between the coefficients (of like terms) of the quadratics for the 3 basic alloys. The calculated mixing fractions are never negative and the linear combination never requires extrapolation because the desired alloy composition point is always inside the triangle with corners at the 3 database alloy points. If the desired alloy composition point were outside this triangle, the linear combination would involve negative mixing fractions and extrapolation which are avoided entirely in the present method.

A different method of interpolation is used in region 3 (Figure 10.3.55) in order to employ the Handbook U-Zr binary correlation for composition points on the side joining corners 2 and 3, i.e., for binary compositions containing 1.5 to 20 wt % Zr. This is done because the Handbook binary correlation Eq. (10.7-12) with Pu content set to zero) is accurate for Zr content in this range, as discussed in Section 10.7.1. The thermal

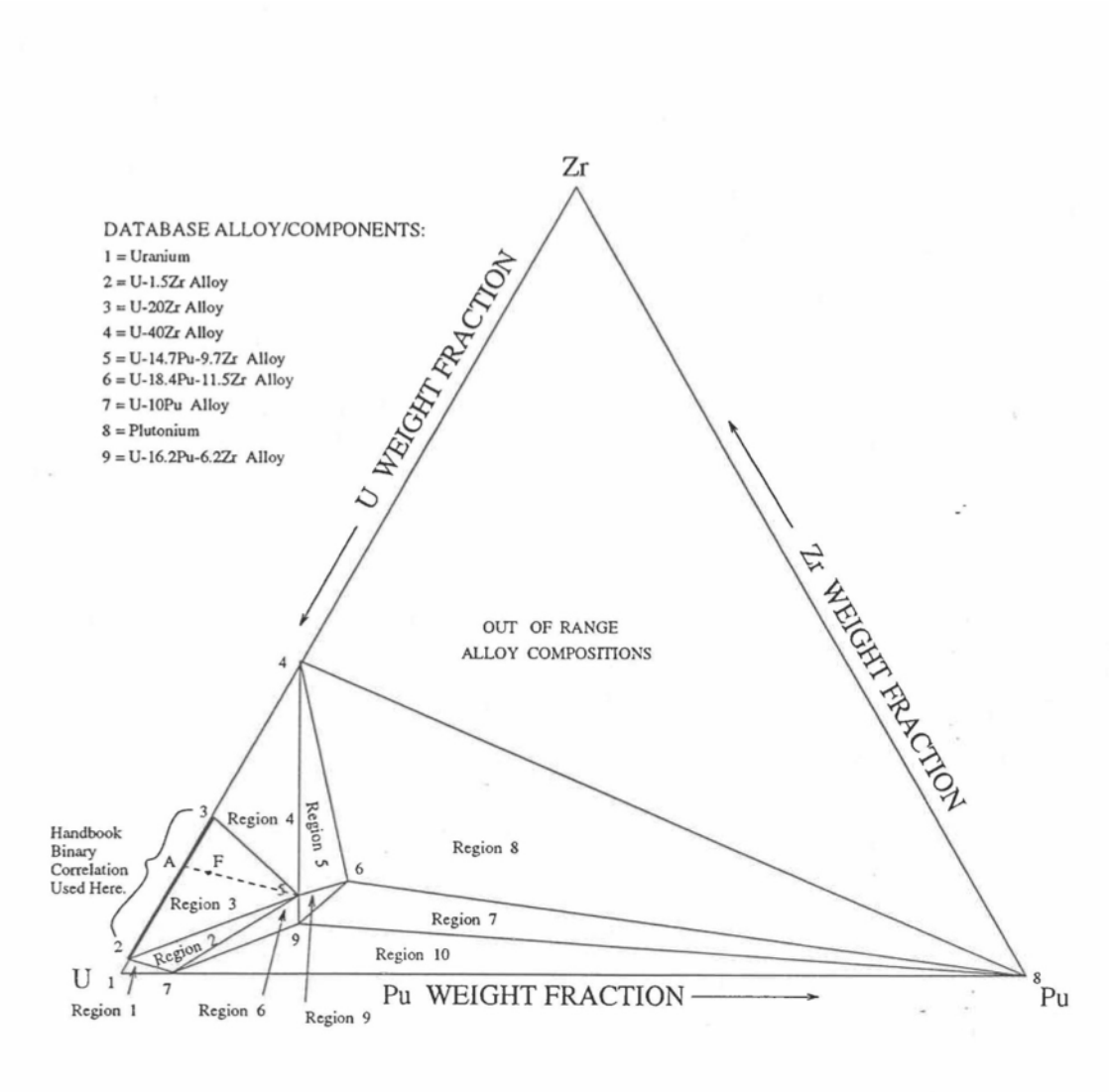


Figure 10.3.55: Diagram Showing Thermal Conductivity Interpolation Regions for 100% Dense Solid U-Pu-Zr Fuels.

conductivity of a desired U-Pu-Zr alloy in region 3 (represented by point F in Figure 10.3.55) is interpolated between only two compositions, the database alloy 5 and binary alloy represented by point A. The point A is the intersection of the side 23 with the line F5 that joins the desired alloy and the database alloy 5. Here also, the interpolation between the database alloy 5 and the binary correlation is performed in terms of the coefficients (of like terms) of the quadratics for thermal conductivity.

10.3.2.5.3 Thermal Conductivity Data of Basic Alloys

Table 10.3.4 gives the thermal conductivity data of the 9 basic solid alloys or components that are used in the present method of evaluating composition- and temperature- dependent thermal conductivity of 100% dense solid U-Pu-Zr fuels. Table 10.3.5 gives the coefficient of the quadratic functions of temperature fitted to the data given in Table 10.3.4. The 9 basic alloys/components are: (1) U metal, (2) U-1.5Zr, (3) U-20Zr, (4) U-40Zr, (5) U-14.7Pu-9.7Zr, (6) U-18.4Pu-11.5Zr, (7) U-10Pu, (8) Pu metal, and (9) U-16.2Pu-6.2Zr. The coefficients of the quadratics for all database alloys except U-1.5 Zr and U-20 Zr are obtained by the method of least-squares fit. The coefficients of the quadratics for U-1.5 Zr an U-20 Zr are evaluated form the Handbook binary correlation. This is required to make the thermal conductivity continuous at the boundary lines of interpolation region 3 (Figure 10.3.55). All the data available in the Metallic Fuels Handbook [10-8, 10-9] have been included in the database used in the present method. Data for two non-Handbook alloys/components, i.e., U-10 Pu data obtained from Billone [10-21] and Pu data reported in the Plutonium handbook [10-23], have also been included in the database of the present method. The U-Pu data is the only intermediate point on the U-Pu binary side of the ternary composition triangle (Figure 10.3.55), and has been included to account for thermal conductivity variation of U-Pu binary as Pu content increases from zero to 100 wt. %. The Pu data has been included to be able to interpolate Mark-V fuel (U-10Pu-10Zr) thermal conductivity because all the other database alloys/components contain les than 20 wt. % Pu. Without the Pu data, the evaluation of thermal conductivity of fuels containing more than 18.4 wt. % Pu will involve extrapolation, which is avoided in the present method.

The quadratic functions given in Table 10.3.5 are used in the temperature range of 293 K to the solidus temperature of the desired U-Pu-Zr alloy. Since the thermal conductivity data of the 9 basic alloys/components (Table 10.3.4) used in the present method go only up to 1173 K, a warning message is printed by the code when the method is used to calculate thermal conductivity at temperatures above 1200 K. A similar warning message is also printed when the method is used at temperatures below 293 K. The thermal conductivity at temperatures above the desired alloys solidus temperature is assumed to be equal to the value at the solidus temperature as discussed in Section E.

Table 10.3.4: Thermal Conductivity Data of the 9 100%-Dense Basic Solid Alloys/Components Used in the Method of Regionwise Interpolation

1 Uranium		2 U-1.5Zr		3 U-20Zr		4 U-40Zr		5 U-14.7Pu-9.7Zr	
Temp (K)	Cond (W/m-K)	Temp (K)	Cond (W/m-K)	Temp (K)	Cond (W/m-K)	Temp (K)	Cond (W/m-K)	Temp (K)	Cond (W/m-K)
293	27	293	22.6	293	11	293	7	373	10
373	29.1	373	24	373	13	373	8	473	13.2
473	31.1	473	26	473	15	473	10	573	16.4
573	33.4	573	38.5	573	17	573	12	673	19.6
673	35.8	673	31	673	20	673	14	773	22.5
773	38.2	773	34	773	22	773	17	873	25.3
873	40.6	873	37	873	25	873	20	973	27.8
973	43.2	973	40.5	973	28	973	24	1073	30.1
1073	45.7	1073	44.5	1073	31	1073	28	1173	32.3
1173	48.3			1173	34	1173	33		

6 U-18.4Pu-11.5Zr		7 U-10Pu		8 Plutonium		9 U-16.2Pu-6.2Zr	
Temp (K)	Cond (W/m-K)	Temp (K)	Cond (W/m-K)	Temp (K)	Cond (W/m-K)	Temp (K)	Cond (W/m-K)
373	9.2	473	23.9	293	$4.1 \pm 6.1\%$	373	10.9
473	11.7	673	28.9	400	$10.2 \pm 9.2\%$	473	14.2
573	14.4	873	36.4	500	$12.3 \pm 6.5\%$	573	17.7
673	17.1	1073	41.9	600	$15.8 \pm 5.4\%$	673	21
773	19.7			700	$18.2 \pm 4.4\%$	773	24.1
873	21.9			800	$18.2 \pm 2.5\%$	873	26.4
973	23.4			900	$20.8 \pm 2.5\%$	973	28
1073	24.8			2000[a]	47.1	1073	29.3
1173	26					1173	30.1

Table 10.3.5: Coefficients of Quadratic Functions of Temperature Fitted to Thermal Conductivity Data of the 9 100%-Dense Basic Solid Alloys/Components Used in the Method of Regionwise Interpolation

No.	Database Alloy/Component	A_{i1} [a]	A_{i2}	A_{i3}	RMS[b] Error, %	Maximum Error, %
1	Uranium	20.931	0.020407	2.4968×10^{-6}	0.30	0.72
2	U-1.5Zr,c	16.516	0.015051	9.3800×10^{-6}	1.94	3.85
3	U-20Zr,c	7.336	0.011791	9.3800×10^{-6}	2.07	5.40
4	U-40Zr	6.0734	-0.0025711	2.1589×10^{-5}	1.78	3.13
5	U-14.7Pu-9.7Zr	-4.3462	0.041352	-8.5931×10^{-6}	0.58	1.18
6	U-18.4Pu011.5Zr	-4.3270	0.040293	-1.2208×10^{-5}	1.71	2.57
7	U-10Pu	10.716	0.025919	3.1250×10^{-6}	1.83	2.34
8	Plutonium	-4.6825	0.033747	-3.9751×10^{-6}	14.6	19.9
9	U-16.2Pu-6.2Zr	-7.6844	0.057035	-2.1028×10^{-5}	1.69	2.74

[a]Thermal conductivity $K_{oi} = A_{i1} + A_{i2}T + A_{i3}T_2$ where i is database alloy number.

[b]RMS Error is root mean square error or the standard deviation.

[c]The coefficients of the quadratic function for these two database alloys are obtained from Handbook binary correlation whereas the coefficients for all the other database alloys are found by least-squares fit.

10.3.2.5.4 Method of Interpolation

The method of interpolation to evaluate the thermal conductivity of a 100%-dense U-Pu-Zr fuel composition at and below its solidus temperature consists of first determining the region of the ternary composition triangle (shown in Figure 10.3.55) in which the desired fuel composition lies, and then finding the weight fractions in which the 3 database alloys of the region are required to be mixed to obtain the desired fuel composition. From the mass balance of U, Pu and Zr, the following equations have been obtained in Section 10.3.2.1 for the weight fractions, X_i, of the database alloys that are assumed to be numbered 1, 2 and 3.

$$X_i = \frac{D_i}{D}, \quad i = 1, 2, 3 \tag{10.3-53}$$

$$D = (W_{z1} - W_{z2})(W_{p3} - W_{p1}) - (W_{p1} - W_{p2})(W_{z3} - W_{z1}) \tag{10.3-54}$$

$$D_1 = (W_z - W_{z2})(W_{p3} - W_p) - (W_p - W_{p2})(W_{z3} - W_z) \tag{10.3-55}$$

$$D_2 = (W_{z1} - W_z)(W_{p3} - W_{p1}) - (W_{p1} - W_p)(W_{z3} - W_{z1}) \tag{10.3-56}$$

$$D_3 = (W_{z1} - W_{z2})(W_p - W_{p1}) - (W_{p1} - W_{p2})(W_z - W_{z1}) \tag{10.3-57}$$

where

W_{ui}, W_{pi}, W_{zi} = weight fractions of U, Pu and Zr in database alloy I of the region in which the desired fuel composition lies,

W_u, W_p, W_z = weight fractions of U, Pu and Zr in the desired fuel composition.

The sum of X_1, X_2 and X_3 is always 1. In case of interpolation, the values of X_1, X_2 and X_3 are always positive. If any extrapolation were involved, the value of a least one weight fraction would be negative which

is entirely avoided in the present method. The coefficients of the Quadratic for the thermal conductivity of the desired 100%-dense U-Pu-Zr fuel composition are then evaluated by mass-weighted averaging of the coefficients (of like terms) of the quadratics for the thermal conductivities of the 3 database alloys.

$$A_{\mathrm{f}1} = X_1 A_{11} + X_2 A_{21} + X_3 A_{31} \tag{10.3-58}$$

$$A_{\mathrm{f}2} = X_1 A_{12} + X_2 A_{22} + X_3 A_{32} \tag{10.3-59}$$

$$A_{\mathrm{f}3} = X_1 A_{13} + X_2 A_{23} + X_3 A_{33} \tag{10.3-60}$$

here

$A_{\mathrm{f}1}, A_{\mathrm{f}2}, A_{\mathrm{f}3}$ = coefficients of the quadratic for thermal conductivity of the desired fuel composition,

$A_{\mathrm{i}1}, A_{\mathrm{i}2}, A_{\mathrm{i}3}$ = coefficients of the quadratic for thermal conductivity of the ith database alloy/component.

The thermal conductivity of the desired 100%-dense U-Pu-Zr fuel composition at a temperature T at or below its solidus is then calculated using its quadratic.

$$K_0(T) = A_{\mathrm{f}1} + A_{\mathrm{f}2} T + A_{\mathrm{f}3} T^2 \tag{10.3-61}$$

where

$K_0(T)$ = thermal conductivity of the desired 100%-dense U-Pu-Zr fuel composition.

It should be noted that the method of interpolation given by Eq. (10.3-58) to Eq. (10.3-61), in which first the coefficients of the quadratic for the desired alloy are calculated and then the thermal conductivity, is equivalent to interpolating the desired alloy thermal conductivity directly from the thermal conductivities of the database alloys.

$$K_0(T) = X_1 K_{01}(T) + X_2 K_{02}(T) + X_3 K_{03}(T) \tag{10.3-62}$$

where

$K_{0\mathrm{i}}(T)$ = thermal conductivity of 100%-dense solid database alloy I of the region in which the desired fuel composition lies.

As described in Section B, the thermal conductivity of a desired U-Pu-Zr fuel composition in region 3, represented by point F (Figure 10.3.55), is interpolated between only tow compositions, the database alloy 5 and the U-Zr binary alloy represented by point A. The point A is the interaction of the side 23 (of region 3) with the line F5 that joins the desired alloy and the database alloy 5. The Zr content of the binary alloy A is given by

$$W_{\mathrm{u}1} X_1 + W_{\mathrm{u}2} X_2 + W_{\mathrm{u}3} X_3 = W_{\mathrm{u}} \tag{10.3-63}$$

where

W_{za} = weight fraction of Zr in the binary alloy A,

$W_{\mathrm{p}5}, W_{\mathrm{z}5}$ = weight fractions of Pu and Zr in the database alloy 5,

$W_{\mathrm{p}}, W_{\mathrm{z}}$ = weight fractions of Pu and Zr in the desired fuel composition.

The coefficients of the quadratic for thermal conductivity of the U-Zr alloy A are found from the IFR Handbook binary correlation [10-8, 10-9], by setting the Pu weight fraction to zero and the Zr weight fraction to W_{za} in Eq. (10.7-12).

$$A_{\mathrm{a}1} = 17.5 \frac{(1 - 2.23 W_{\mathrm{za}})}{(1 + 1.61 W_{\mathrm{za}})} \tag{10.3-64}$$

$$A_{\mathrm{a}2} = 0.0154 \frac{(1 + 0.061 W_{\mathrm{za}})}{(1 + 1.61 W_{\mathrm{za}})} \tag{10.3-65}$$

$$A_{a3} = 9.38 \times 10^{-6} \tag{10.3-66}$$

where

A_{a1}, A_{a2}, A_{a3} = coefficients of the quadratic for thermal conductivity of binary alloy A.

The coefficients of the quadratic for thermal conductivity of the desired fuel composition in region 3 are then interpolated between the coefficients of the quadratic for binary alloy A and those for database alloy 5.

$$A_{fj} = A_{aj} + (A_{5j} - A_{aj}) \frac{W_p}{W_{p5}}, \quad j = 1, 2, 3 \tag{10.3-67}$$

where

A_{f1}, A_{f2}, A_{f3} = coefficient of the quadratic for thermal conductivity of the desired fuel represented by point F in region 3.

10.3.2.5.5 Thermal Conductivity During and After Melting

The quadratic functions of temperature, given in Table 10.3.5, are used to interpolate thermal conductivity of a desired U-Pu-Zr alloy in the temperature range of 293 K to the solidus temperature of the desired alloy. The thermal conductivity at temperatures above the desired alloy solidus temperature is assumed to be equal to the value at the solidus temperature because of two reasons: (1) U-Pu-Zr alloy thermal conductivity data during and after melting are not available, and (2) molten uranium thermal conductivity estimated form its electrical resistivity [10-24] using Wiedermann-Lorenz law, as described below, is nearly equal to solid uranium thermal conductivity evaluated at its melting point using the quadratic function given in Table 10.3.5.

Wiedermann-Lorenz law [10-25] gives the following relation between thermal conductivity and electrical resistivity of metals and alloys at temperatures above 150 K.

$$\frac{\rho K_o}{T} = L \tag{10.3-68}$$

where

K_o = thermal conductivity of 100% dense metal or alloy, W/m-K,

ρ = electrical resistivity of the material, Ω-m,

L = a constant called Lorenz number determined from theoretical considerations to be 2.45×10^{-4} W-Ω/K^2 for all metals.

Table 10.3.6 gives the electrical resistivity data [10-24] for uranium at temperatures from 293 K to 1573 K. The thermal conductivities at lower (than melting) temperatures, i.e., 293 K, 873 K, 973 K and 1073 K, are also included in Table 10.3.6 to examine the difference between the experimentally determined Lorenz numbers and the theoretical value of 2.45×10^{-8} W-Ω/K^2. The Lorentz numbers determined from data at these four lower (than melting) temperatures are approximately equal to the theoretical value. The thermal conductivities of uranium at and above the melting temperature have been calculated using the theoretical value of the Lorenz number, and found to be 54.2, 55.3 and 56.8 W/m-K at 1406 K, 1473 K and 1573 K. These thermal conductivity estimates are nearly equal to the value of 54.6 W/m-K calculated for solid uranium at its melting point (of 1408 K that is used in SAS4A/SASSYS-1 codes) using the quadratic function given in Table 10.3.5. From this comparison, it is concluded that there is no discontinuity in uranium thermal conductivity at its melting point. It should be noted from plots [10-18] of thermal conductivity as a function of temperature that several other metals do show a sharp discontinuity at the melting point.

At present, the thermal conductivity of all alloy compositions at temperatures above the solidus are assumed to remain constant and equal to the value at the solidus, and a warning message is printed by the code whenever this assumption is used.

10.3.2.5.6 Subroutine KFUEL1 and Its Comparison With the Handbook Data

Figure 10.3.56 shows flow diagram of the subroutine KFUEL1 for calculating the thermal conductivity, based on this method of regionwise interpolation, of a 100% dense U-Pu-Zr fuel as a function of composition and temperature. This subroutine has been incorporated into the SAS4A/SASSYS-1 codes. For improving the computational speed, the coefficients of the quadratic for thermal conductivity of 100% dense U-Pu-Zr alloy of each input fuel type is pre-calculated before steady-state calculation, and used later in the steady-state and transient calculations of the codes.

Table 10.3.6: Estimation of 100%-Dense Molten Uranium Thermal Conductivity Using Wiedermann-Lorenz Law

Temperature, K	Electrical Resistivity, Ω-m	Thermal Conductivity,[a] W/m-K	Computed Value of L, W-Ω/K^2
293	30.0×10^{-8}	27.0	2.76×10^{-8}
873	59.0×10^{-8}	40.6	2.74×10^{-8}
973	55.5×10^{-8}	43.2	2.46×10^{-8}
1073	54.0×10^{-8}	45.7	2.30×10^{-8}
1406[b]	63.6×10^{-8}	54.2	2.45×10^{-8}
1473	65.3×10^{-8}	55.3	2.45×10^{-8}
1573	67.8×10^{-8}	56.8	2.45×10^{-8}

[a]The first four values are obtained from IFR Handbook data summarized in Table 10.3.4, and the next three values at 1406 K, 1473 K and 1573 K, are calculated using Wiedermann-Lorenz Law with constant $L = 2.45 \times 10^{-8}$ W-Ω/K^2.

[b]Uranium melting point reported in Ref. [10-24] compared to 1408 K given in the IFR Handbook and used in SAS4A/SASSYS-1 codes. The value 1406 K is used only in this table for estimating molten uranium thermal conductivity.

Figure 10.3.57 to Figure 10.3.67 show the comparison of calculated thermal conductivity with all the data in the Metallic Fuels Handbook [10-8] and with the data for U-10Pu alloy [10-21] and Pu metal [10-23] given in Table 10.3.4. There is a close agreement between the calculated and the measured thermal conductivities of all these alloys except U-11.4 Zr and Pu metal. In the case of u-11.4 Zr, there are two outlying data points, one at 673 K and the other at 1173 K, and the rest of the data points are in fair agreement with the calculated curve (Figure 10.3.60). It should be noted that the calculated curve for this alloy basically arises from the Handbook binary correlation [10-8], which is accurate for both the neighboring binary alloys (i.e., U-5Zr and U20Zr plotted in Figure 10.3.59 and Figure 10.3.61) for which data are available. Based on the degree of confidence in the Handbook binary correlation, the two outlying data points at 673 K and 1173 K are not believed to be reliable enough to require a revision of the subroutine KFUEL1. In the case of Pu metal, the scatter of the data points about the calculated curve is caused by a number of solid-state phase transitions. However, the behavior of Pu when alloyed with U and Zr is different from its behavior in isolation, and the smooth quadratic function of temperature use in calculation (in the subroutine KFUEL1) is closer to the expected behavior when alloyed.

Figure 10.3.68 to Figure 10.3.70 show the variation of thermal conductivity with temperature for 20 100%-dense alloy compositions obtained from this subroutine. The Pu content varies from zero to 30 wt. % and the Zr content varies from 1.0 to 30 wt. % in the alloy compositions plotted in the figures. The solidus and liquidus temperatures required as input to the KFUEL1 subroutine were calculated using another subroutine developed by Pelton [10-13]. It should be noted that each thermal conductivity curve is flat at temperatures above the solidus temperature of the alloy. The thermal conductivities do not show any unphysical or unexpected behavior.

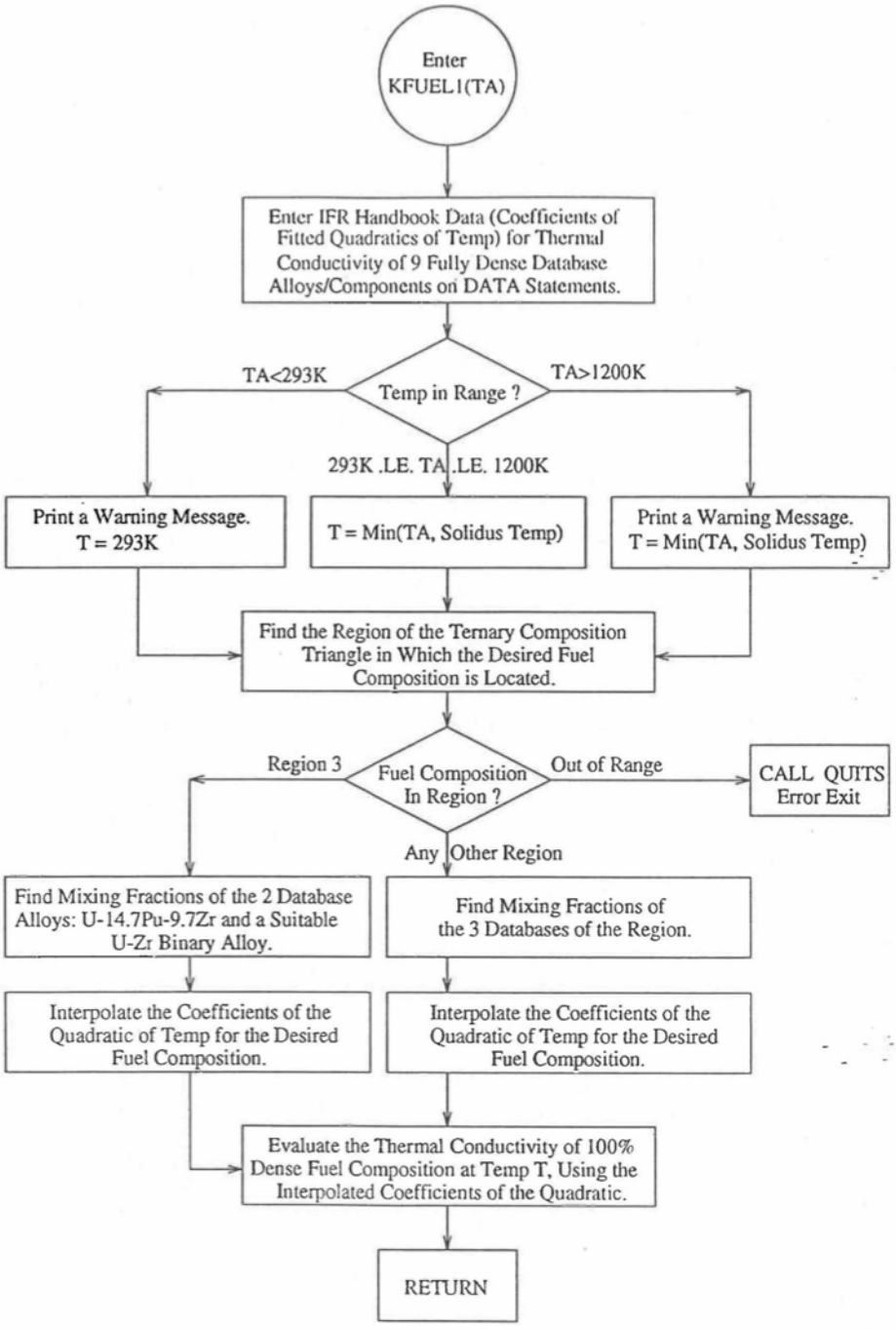


Figure 10.3.56: Flow Diagram of Subroutine KFUEL1 for 100% Dense U-Pu-Zr Thermal Conductivity.

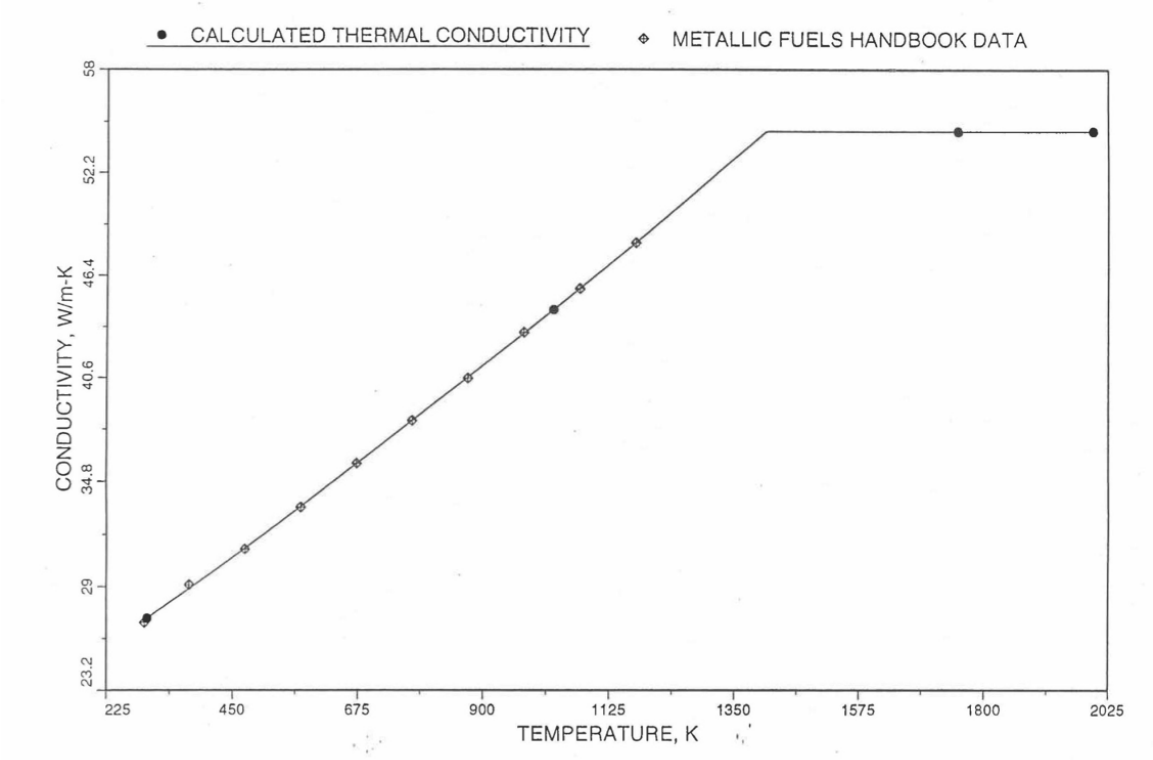


Figure 10.3.57: Comparison of 100% Dense Uranium Metal Thermal Conductivity Data in IFR Handbook With Those Calculated by Subroutine KFUEL1.

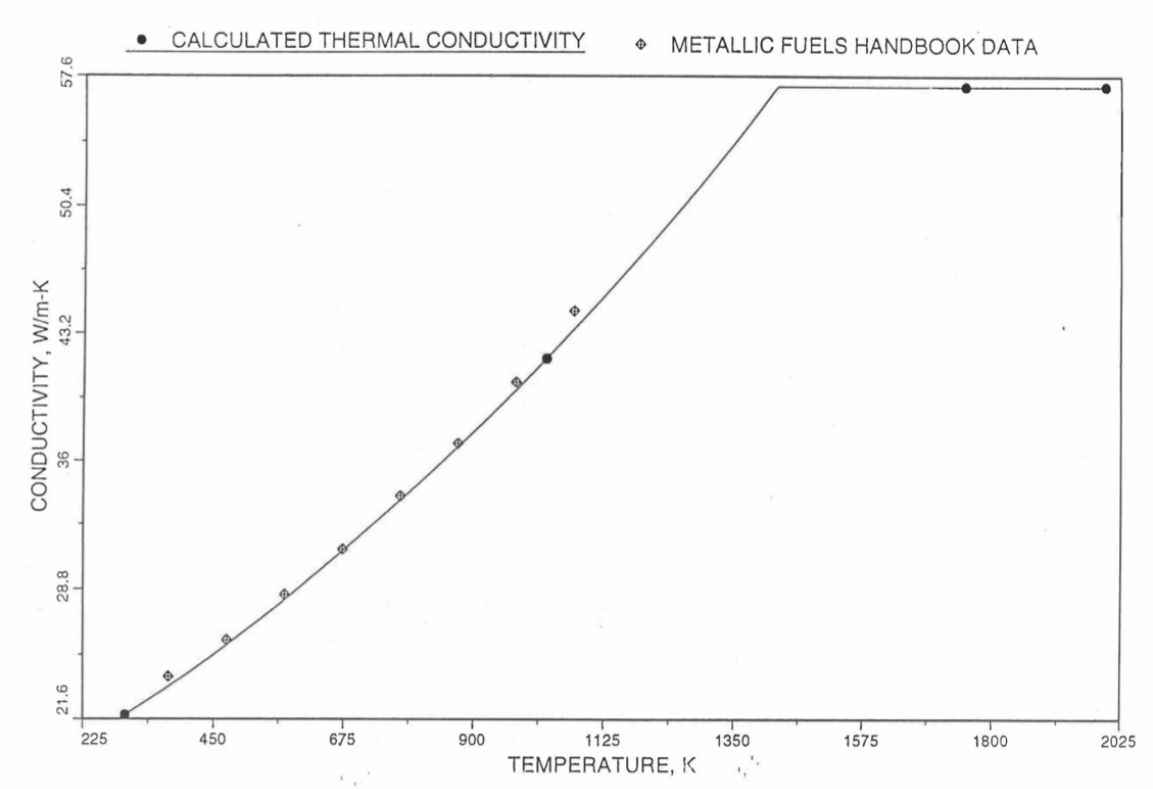


Figure 10.3.58: Comparison of 100% Dense U-1.5Zr Fuel thermal Conductivity Data in IFR Handbook With Those Calculated by Subroutine KFUEL1.

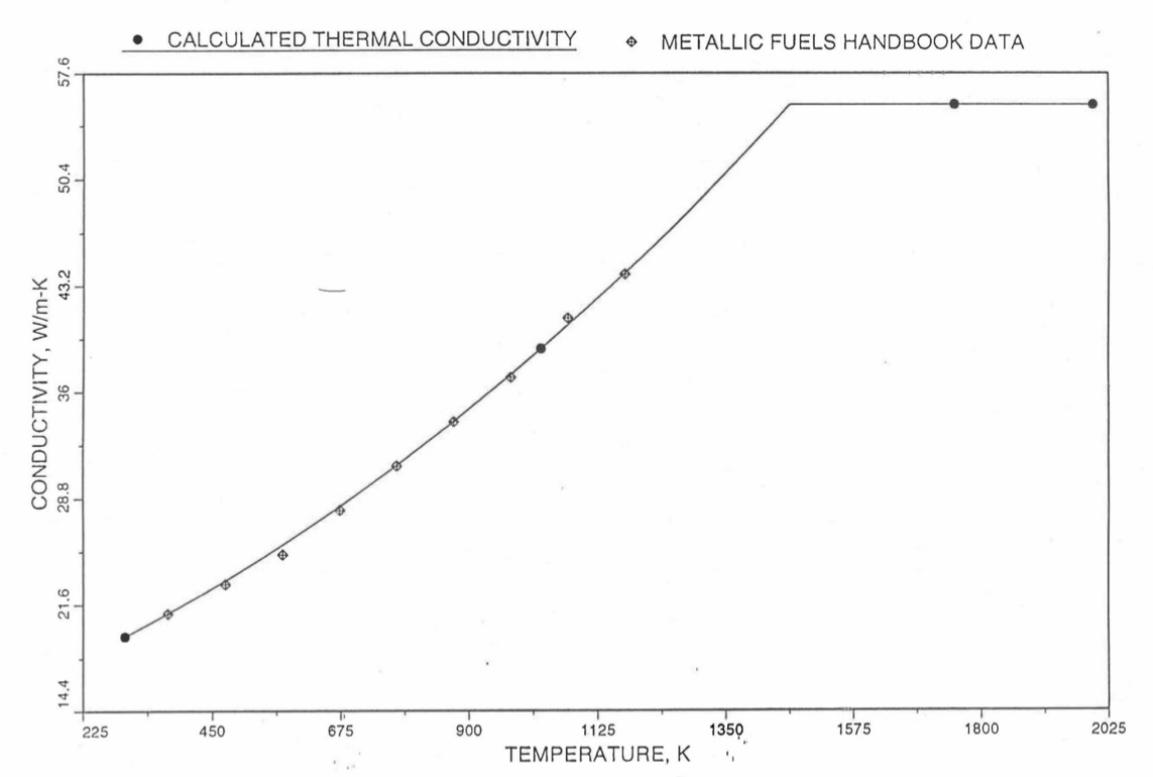


Figure 10.3.59: Comparison of 100% Dense U-5.9Zr Fuel Thermal Conductivity Data in IFR Handbook With Those Calculated by Subroutine KFUEL1.

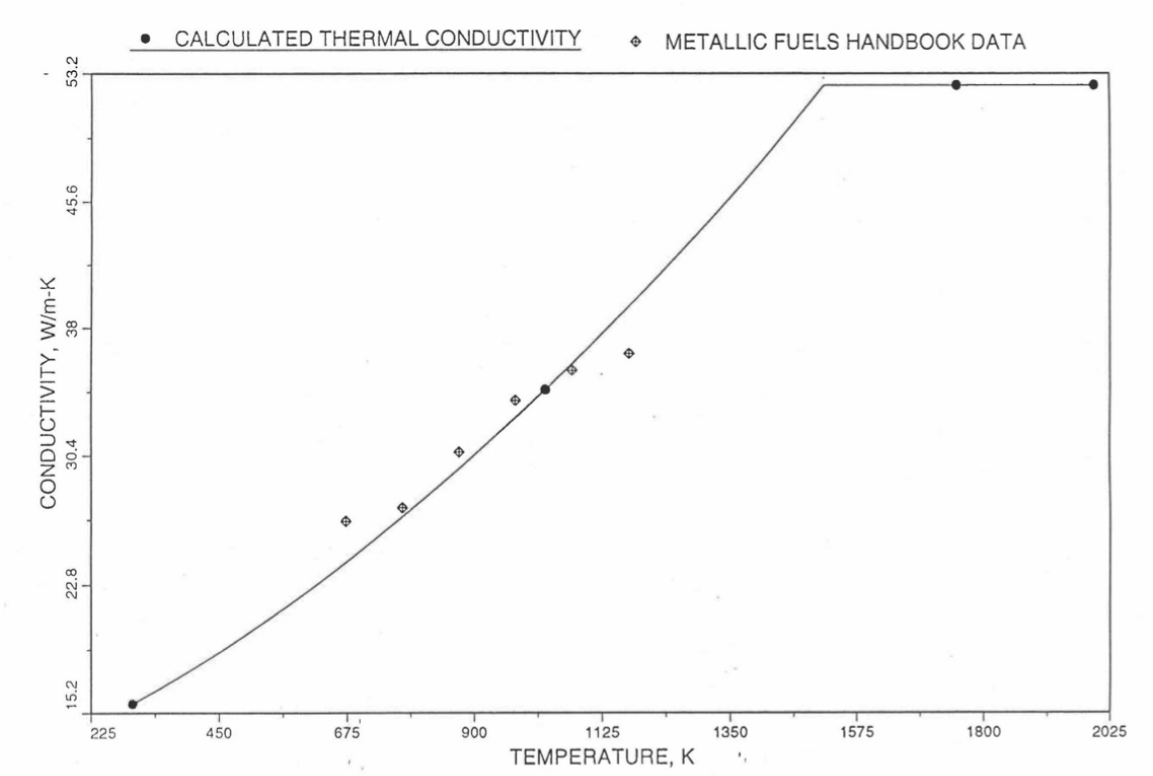


Figure 10.3.60: Comparison of 100% Desense U-11.4Zr Fuel Thermal Conductivity Data in IFR Handbook With Those Calculated by Subroutine KFUEL1.

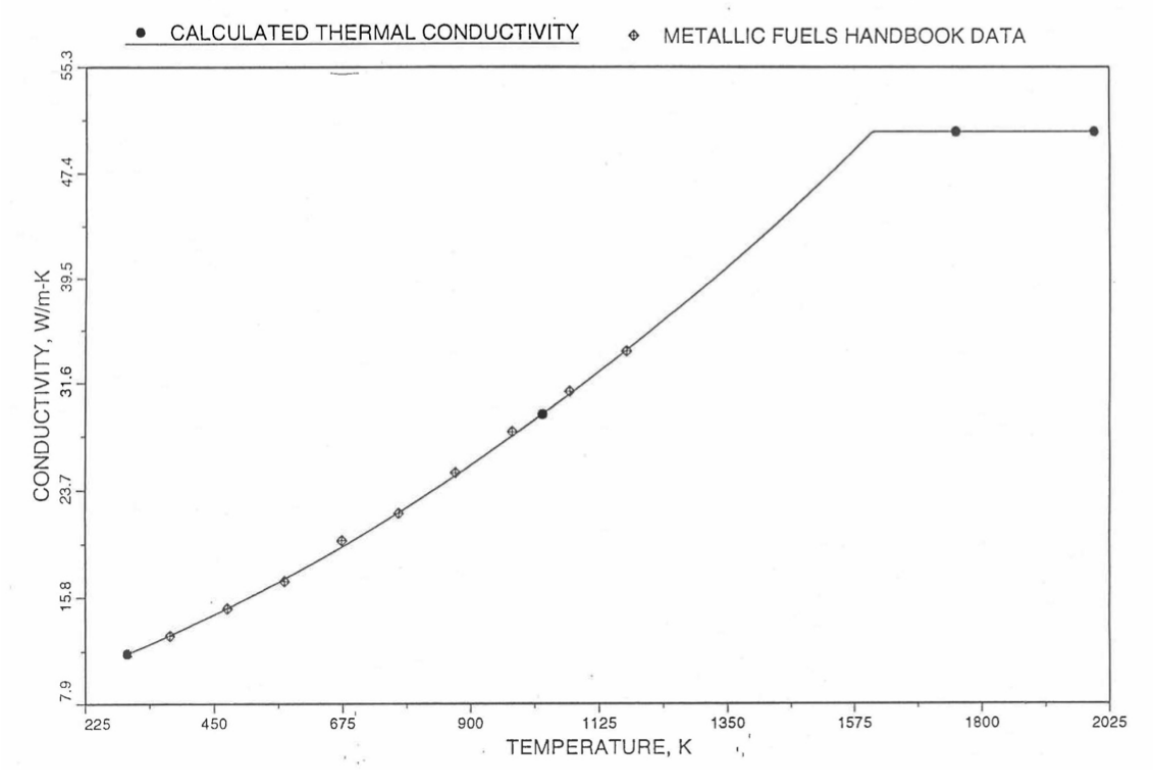


Figure 10.3.61: Comparison of 100% Dense U-20Zr Fuel Thermal Conductivity Data in IFR Handbook With Those Calculated by Subroutine KFUEL1.

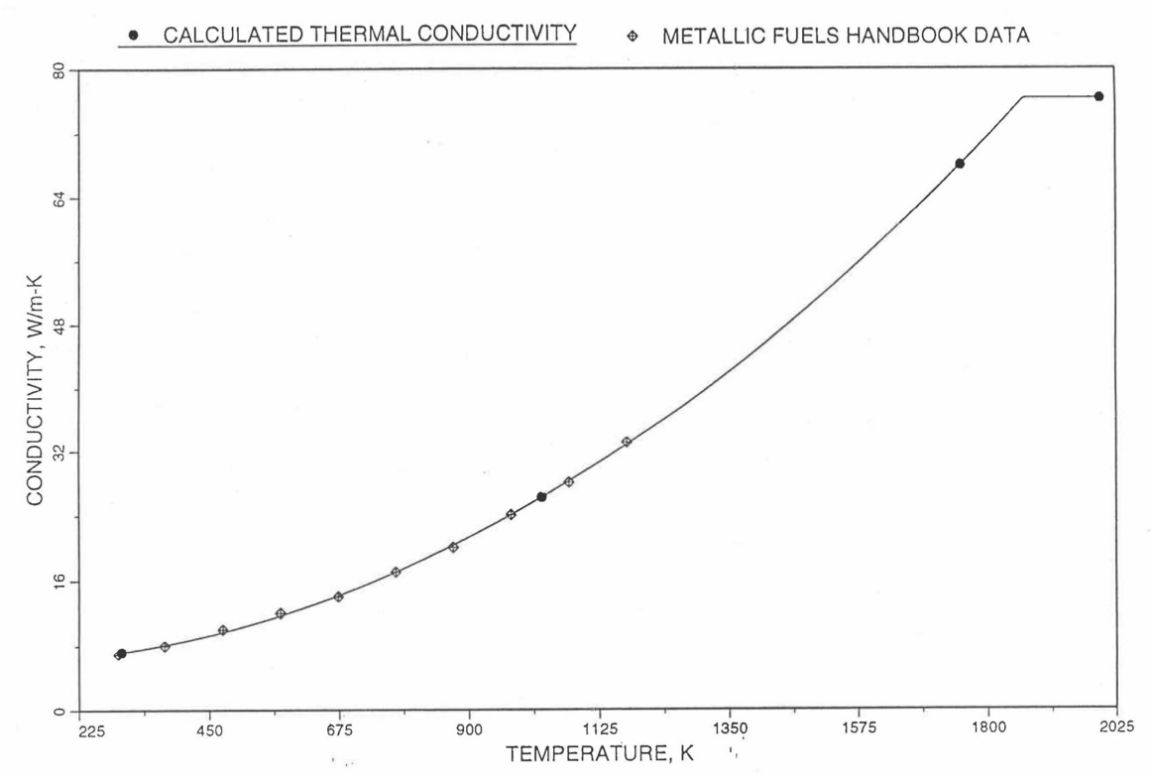


Figure 10.3.62: Comparison of 100% Dense U-40Zr Fuel Thermal Conductivity Data in IFR Handbook With Those Calculated by Subroutine KFUEL1.

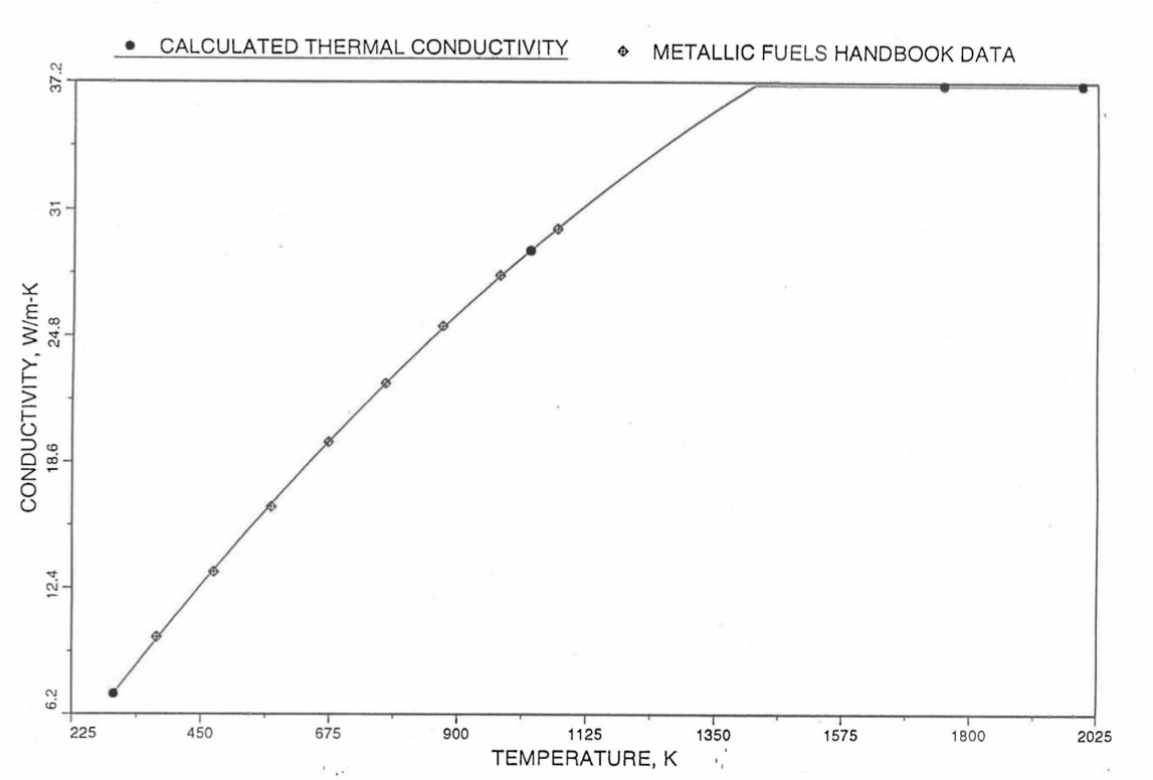


Figure 10.3.63: Comparison of 100% Dense U-14.7Pu-9.7Zr Fuel Thermal Conductivity Data in IFR Handbook With Those Calculated by Subroutine KFUEL1.

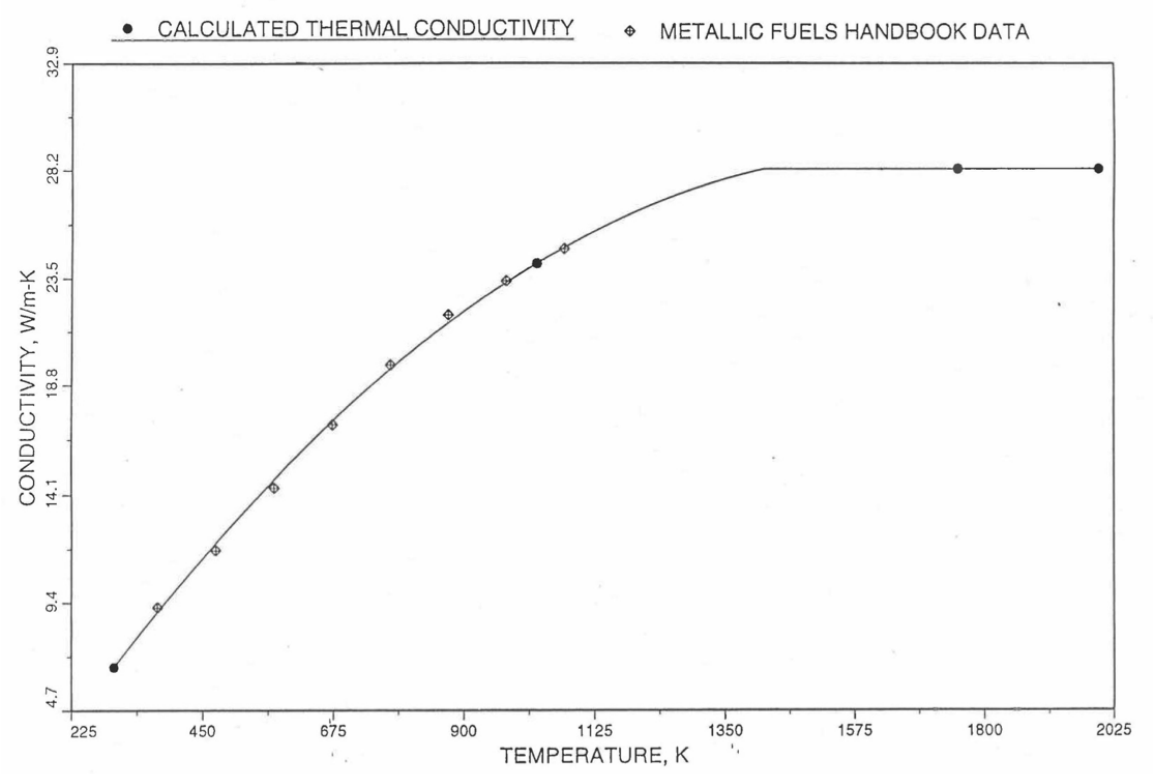


Figure 10.3.64: Comparison of 100% Dense U-18.4Pu-11.5Zr Fuel Thermal Conductivity Data in IFR Handbook With Those Calculated by Subroutine KFUEL1.

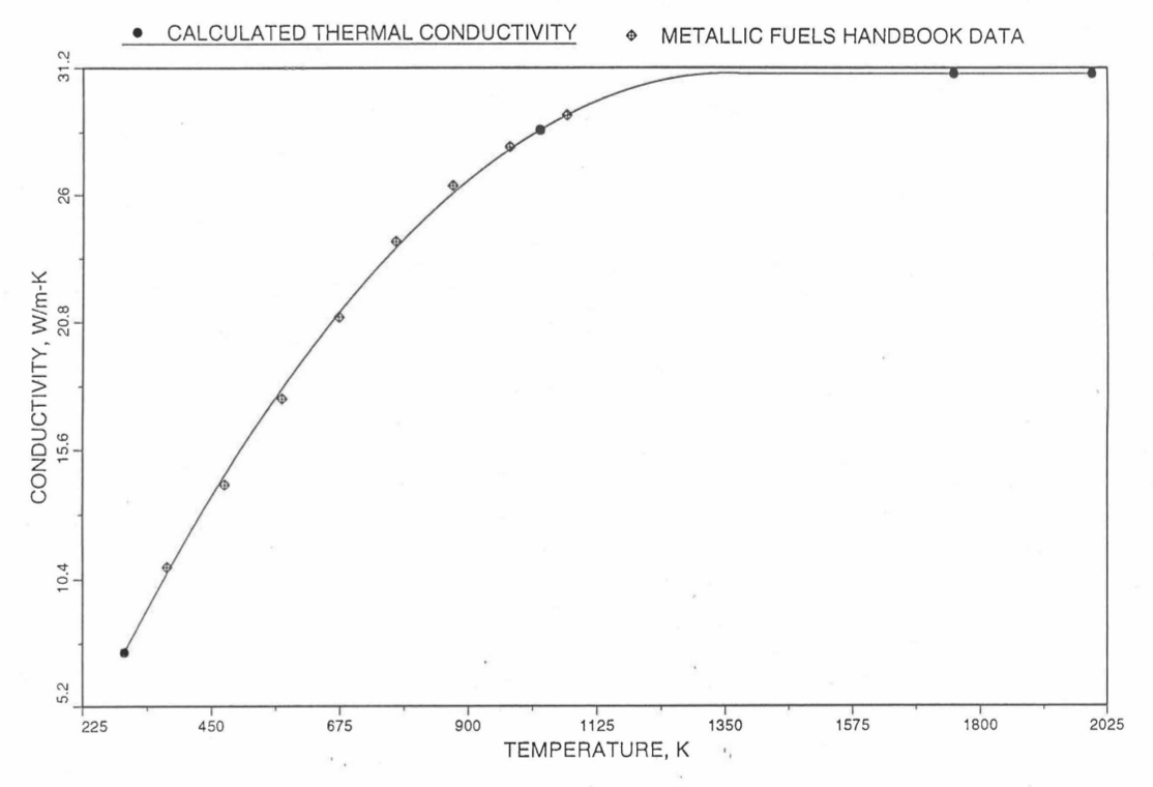


Figure 10.3.65: Comparison of 100% Dense U-16.2Pu-6.2Zr Fuel Thermal Conductivity Data in IFR Handbook With Those Calculated by Subroutine KFUEL1.

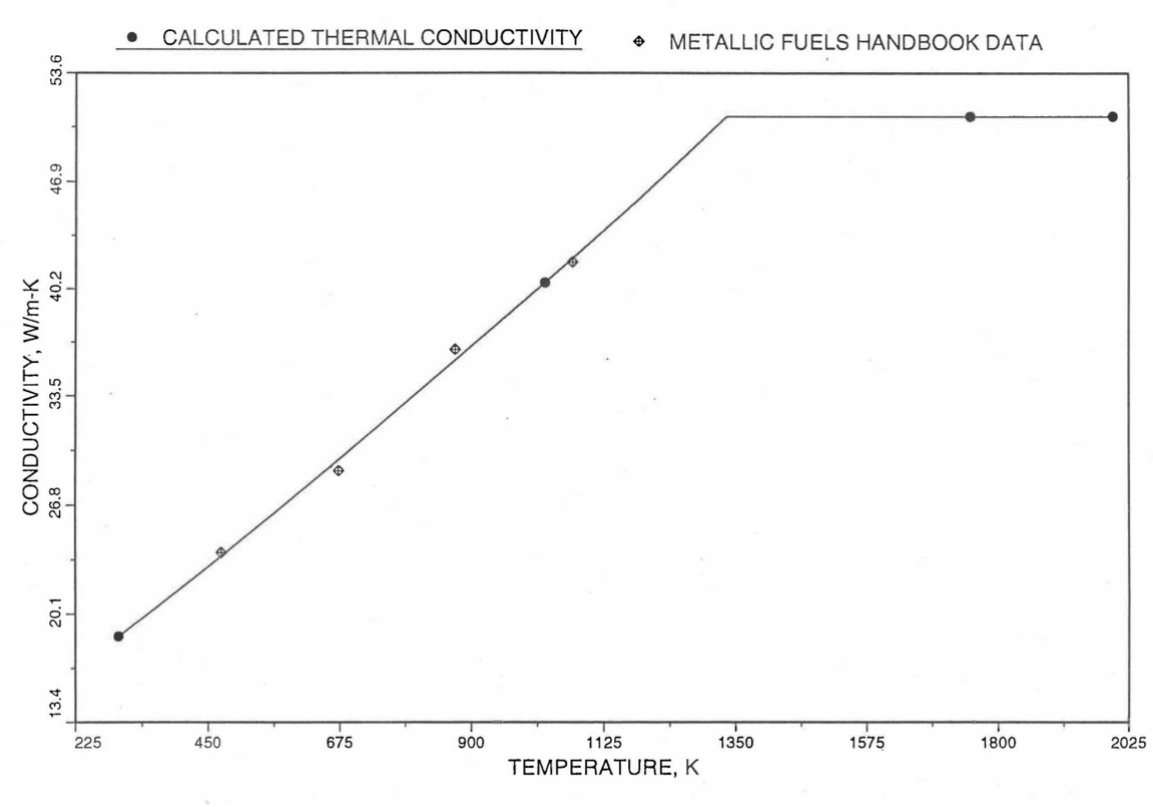


Figure 10.3.66: Comparison of 100% Dense U-10Pu Fuel Thermal Conductivity Data With Those Calculated by Subroutine KFUEL1.

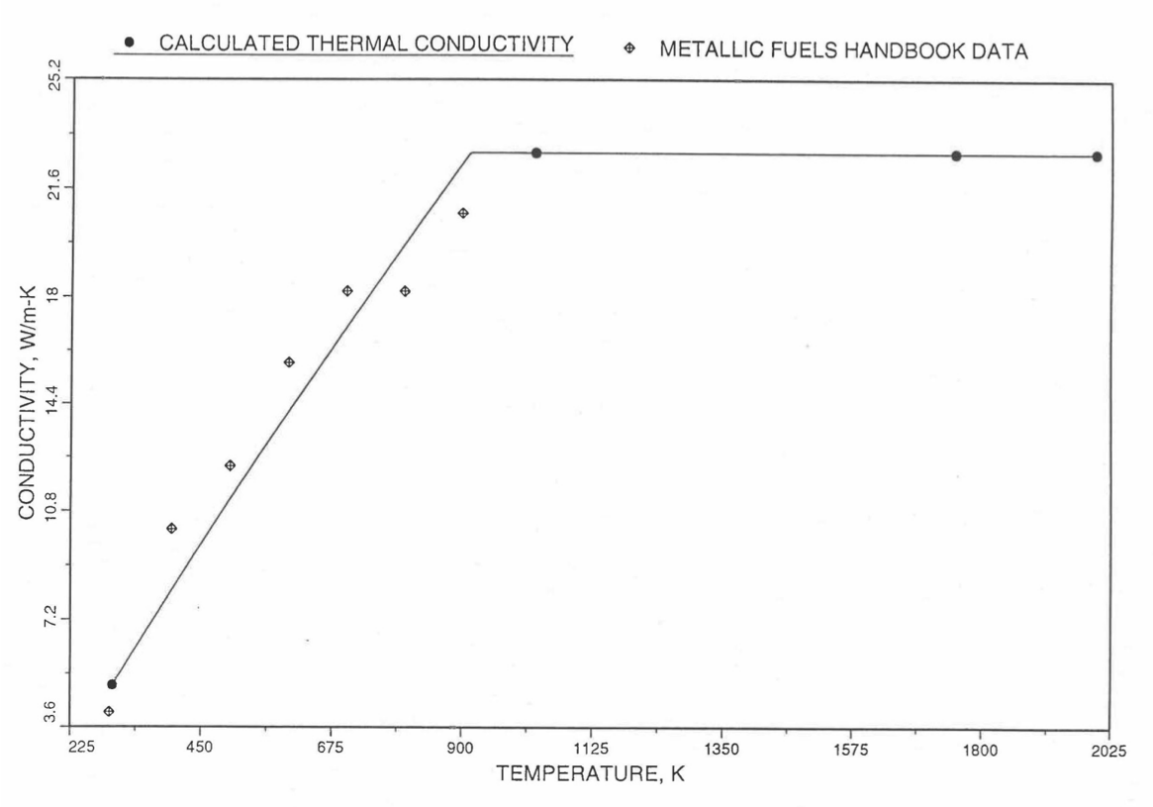


Figure 10.3.67: Comparison of 100% Dense Pu Metal Thermal Conductivity Data in Plutonium Handbook With Those Calculated by Subroutine KFUEL1.

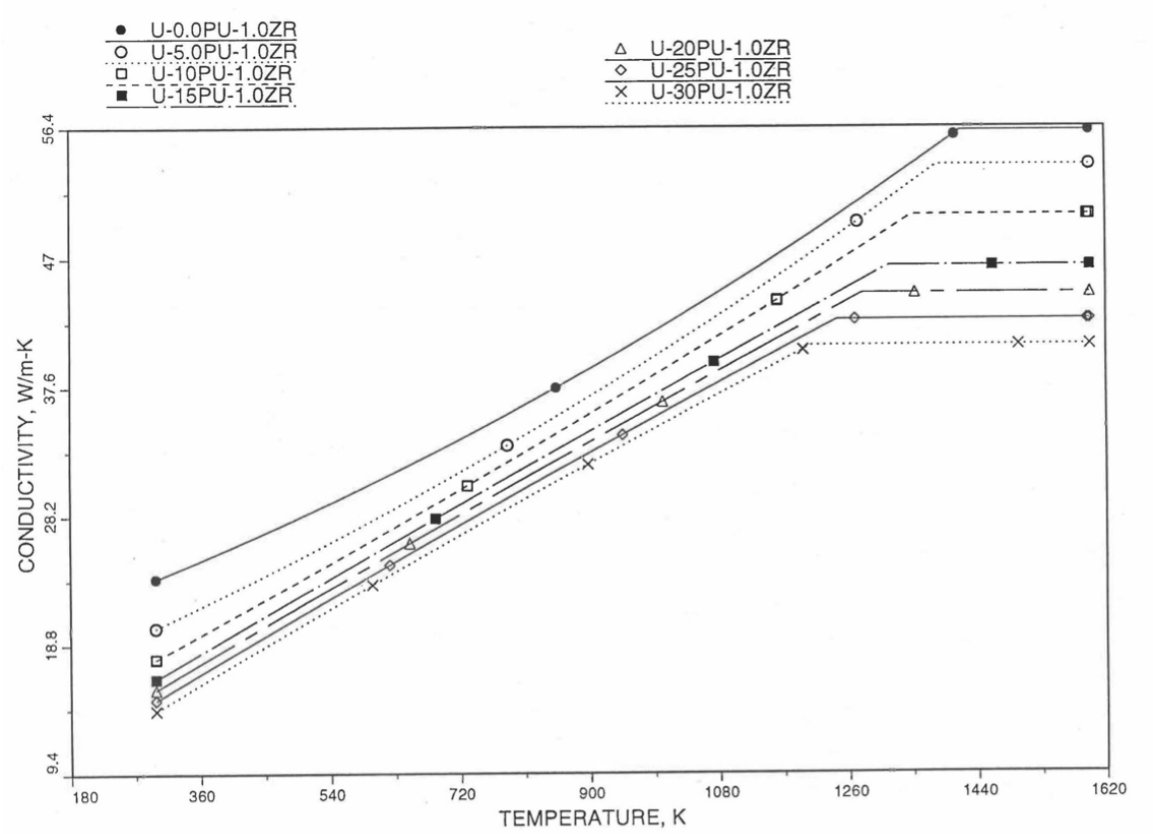


Figure 10.3.68: Thermal Conductivity of 100% Dense U-Pu-1.0Zr Alloy Fuel, Obtained From Regionwise Interpolation of IFR Handbook Data.

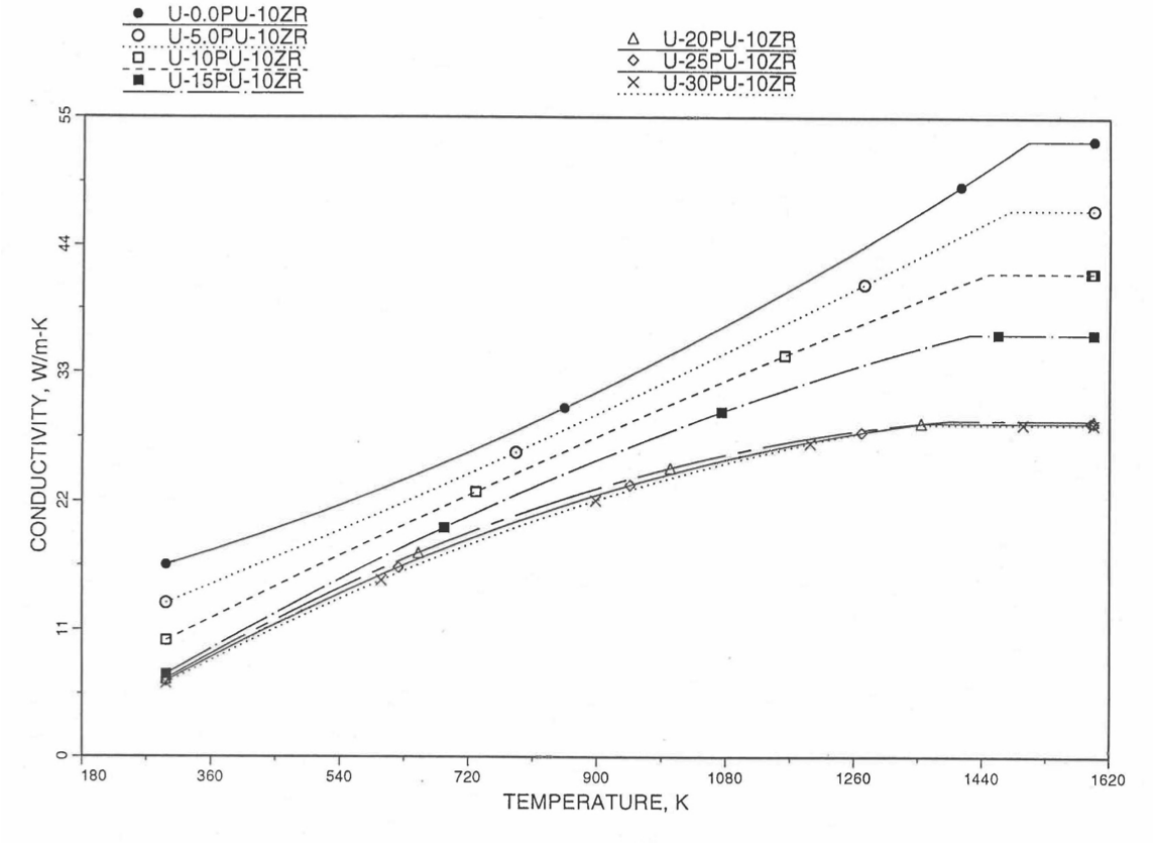


Figure 10.3.69: Thermal Conductivity of 100% Dense U-Pu-10Zr Alloy Fuel, Obtained From Regionwise Interpolation of IFR Handbook Data.

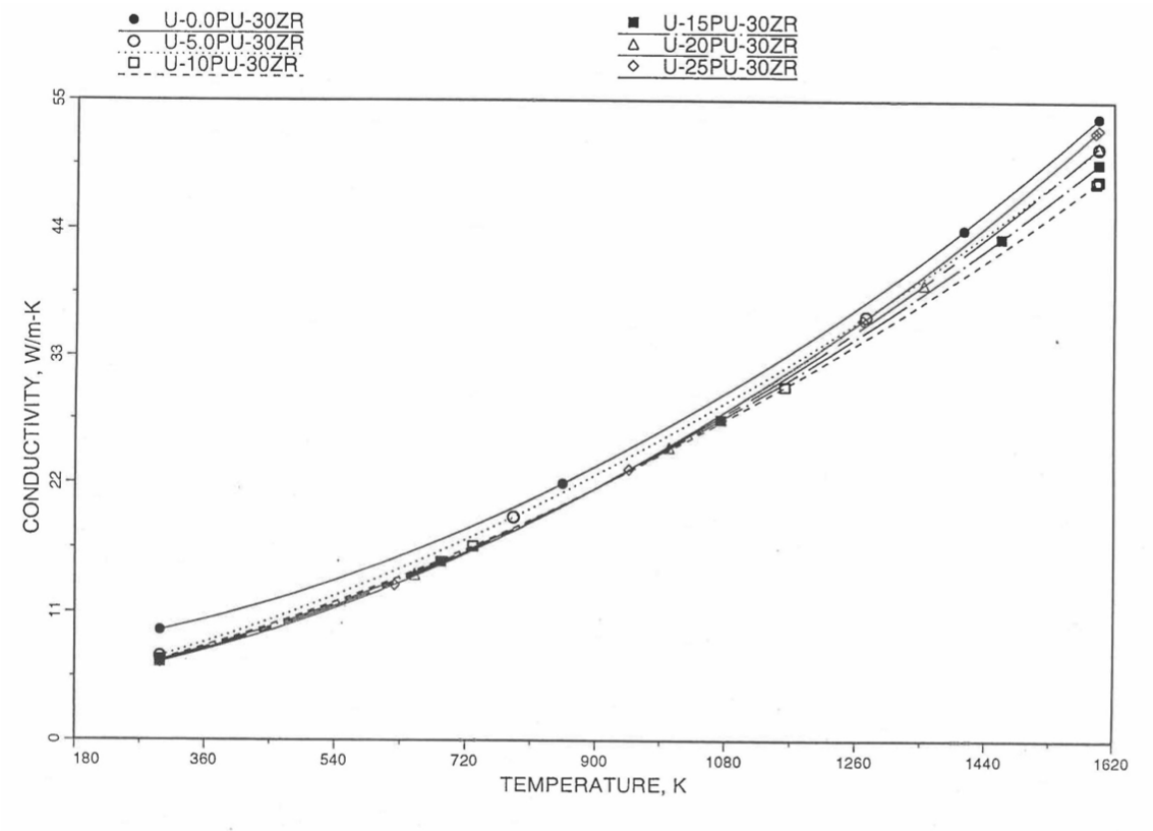


Figure 10.3.70: Thermal Conductivity of 100% Dense U-Pu-30Zr Alloy Fuel, Obtained From Regionwise Interpolation IFR Handbook Data.

10.3.2.6 Burnup or Porosity Correction of Thermal Conductivity

The calculation of thermal conductivity of unirradiated 100% dense U-Pu-Zr alloy fuel as a function of composition and temperature is described in Section 10.3.2.5. The thermal conductivity also depends strongly on the fission gas-filled and logged sodium-filled porosities that form in the fuel during normal steady-state irradiation. This section describes the following two methods that are used in the SAS4A/SASSYS-1 codes to correct the thermal conductivity of 100% dense U-Pu-Zr alloy fuel for fission gas-filled and logged sodium-filled porosities:

(i) an empirical method based on experimental data for U-Fs alloy fuel thermal conductivity change with burnup in EBR-II,

(ii) a theoretical method [10-4, 10-26] that explicitly accounts for the gas-and sodium-filled porosities.

These two options are controlled through the input parameter KFIRR.

10.3.2.6.1 Empirical Correction Based on Average Burnup

No experimental data are available for the effects of irradiation on the thermal conductivity of U-Pu-Zr ternary or U-Zr binary alloy fuel. It is assumed that U-Pu-Zr ternary and U-Zr binary alloy fuels behave during steady-state irradiation in a similar manner to U-5Fs fuel Figure 10.3.68 (for which data re available) with regard to fission gas-induced swelling and sodium logging. The empirical correction factor, i.e., the irradiated-to-unirradiated thermal conductivity ratio, obtained from the analysis of experimental data [10-27 to 10-29] for the effects of irradiation on U-5Fs fuel thermal conductivity, is also used in the Metallic Fuel Handbook [10-8] for the U-Pu-Zr alloy and U-Zr alloy fuels.

A number of irradiation-induced phenomena tend to lower the thermal conductivity of U-5Fs fuel from start-up to about 2 at. % burnup. During this period irradiation, the fuel porosity is not interlinked. Fission gas bubbles and thermal stress-induced microcracking lower the fuel density and thermal conductivity; solid and liquid fission-products buildup tends to lower the thermal conductivity in much the same way as increasing amounts of fissium lower the conductivity; and irradiation damage tends to degrade the thermal conductivity. But the dominant mechanism accounting for the degradation of U-5Fs thermal conductivity from start-up to about 2 at. % burnup is assumed to be the decrease in fuel density due to fission gas-induced swelling and microcracking. Based on the experimental data of diNovi [10-27] and Beck and Fousek [10-28] reviewed in the Metallic Fuels Handbook [10-8], the recommended porosity correction factor, i.e., the irradiated-to-unirradiated thermal conductivity ratio is

$$\frac{K}{K_{\mathrm{o}}} = \frac{1 - P'}{1 + 1.7P'} \tag{10.3-69}$$

where

P' = porosity fraction relative to the initial 100% dense fuel volume,

K = thermal conductivity of irradiated fuel,

K_{o} = thermal conductivity of unirradiated 100% dense fuel.

The interlinkage of fission gas bubbles is assumed to occur at 2 at. % burnup. Beyond 2 at. % burnup, the effective thermal conductivity tends to increase due to sodium logging and fuel hot pressing, and tends to decrease due to solid and liquid fission-products buildup. But the ingress of bond sodium into the bubbles dominates and improves the thermal conductivity somewhat, as suggested by the in-core measurements of Betten [10-29]. He has reported U-5Fs fuel and coolant thermocouple measurements in an instrumented subassembly, as a function of time to a peak subassembly burnup of 8.8 at. %. From startup to 2 at. % burnup, U-5Fs fuel thermal conductivity shows the expected decrease with burnup due to porosity formation, and the irradiated-to-unirradiated thermal conductivity ratio takes a minimum value of 0.475 with a standard deviation of 0.111. Beyond 2 at. % burnup, the thermocouple measurements show an improvement in the effective thermal conductivity, which

levels off, after about 5 at. % burnup, to a steady-state value. The long-time value of U-Fs fuel irradiated-to-unirradiated thermal conductivity ratio is 0.72 with a standard deviation of 0.06.

The minimum value (just before porosity interlinkage at 2 at. % burnup) of irradiated-to-unirradiated thermal conductivity ratio, recommended in the Metallic Fuels Handbook [10-9], for U-Zr binary and U-Pu-Zr ternary fuels is 0.5. The porosity fraction *P'*, corresponding to this value (i.e., 0.5) of the ratio K/K_{o} is obtained from Eq. (10.3-69) to be 0.27. Following a recent recommendation by Billone [10-21] used in Mark-V fuel safety calculations, the porosity fraction is assumed to vary linearly with burnup during early irradiation before porosity interlinkage, equaling 0.27 at 2 at. % burnup.

$$P' = 0.135 \text{ Bu}, \quad \text{Bu } \leq 2 \text{ at. \%}, \tag{10.3-70}$$

where

Bu = fuel burnup, at. %.

The long-time value (after porosity interlinkage, sodium logging and fuel hot pressing at burnups beyond 5 at. %) of irradiated-to-unirradiated thermal conductivity ratio, recommended in the IFR Handbook, for U-Zr and U-Pu-Zr fuels is 0.7.

The above conclusions about the empirical correction of U-Pu-Zr alloy fuel thermal conductivity for the effects of steady-state irradiation can be summarized as follows:

$$\frac{K}{K_{\mathrm{o}}} = \begin{cases} \frac{(1-P')}{(1+1.7P')}, & \text{Bu} \leq 2 \text{ at. \%} \\ 0.5 + 0.0667\,(\text{ Bu} - 2)\,, & 2 < \text{ Bu} \leq 5 \text{ at. \%} \\ 0.7, & \text{Bu} > 5 \text{ at. \%} \end{cases} \tag{10.3-71}$$

10.3.2.6.2 Theoretical Correction Based on Sodium- and Gas-Filled Porosities

The fuel develops fission gas-filled porosities during steady-state operation of the reactor. The bond sodium used in the fuel-cladding gap of U-Pu-Zr alloy fuel pins can also migrate into the fuel and fill some of the porosities of the fuel after porosity interlinkage. The gas-filled porosities tend to decrease the fuel thermal conductivity whereas the sodium-filled porosities tend to increase it. The empirical correction based on average fuel burnup is suitable for transient calculations in which fuel porosity and sodium logging fraction do not change during the transient. If the distributions of fuel porosity and sodium logging fraction change during the transient, a method of correcting fuel thermal conductivity for irradiation effects that explicitly accounts for sodium- and gas-filled porosities is required. In this section a formula is derived for the evaluation of these two effects using a generalization of the method given in Ref. [10-26] for single type of porosity.

Assuming all the gas-filled pores to be cubes of the same size (side a_1) and all the sodium-filled pores to be cubes of another size (side a_2), a unit cell of the porous metallic fuel is constructed that contains a single gas-filled pore and a single sodium-filled pore as shown in Figure 10.3.71. The unit cell is a cube of side L and contains fully dense fuel surrounding the pores. In a real material the pores are not uniformly arranged like atoms in a crystal lattice; so the amount of solid fuel in the unit cell represents an average value. Assuming the heat flow in the y-direction only, the total heat flow through the front x-z face of the unit cell passes through three heat flow paths in parallel: (1) the pore tube containing the gas-filled pore (see Figure 10.3.71), (2) the pore tube containing the sodium-filled pore, and (3) the remaining frontal are of the fully dense fuel of thermal conductivity K_{s}. The effective thermal conductivity of the unit cell in the y-direction is given by

$$K = P_{\mathrm{c1}} K_{\mathrm{pt1}} + P_{\mathrm{c2}} K_{\mathrm{pt2}} + (1 - P_{\mathrm{c1}} - P_{\mathrm{c2}})\, K_{\mathrm{s}} \tag{10.3-72}$$

where

K = effective thermal conductivity of the fuel containing sodium-and gas-filled porosities.

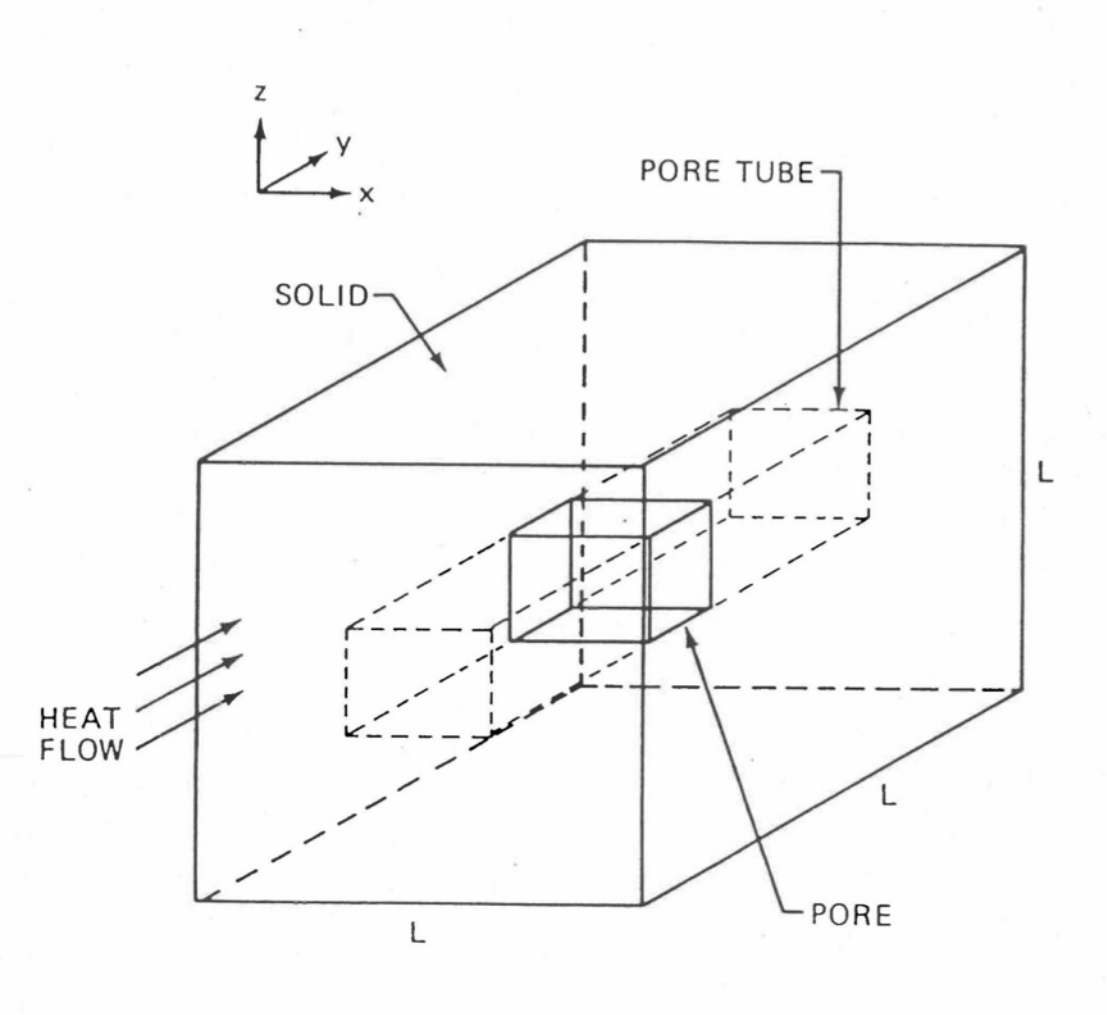


Figure 10.3.71: A Unit Cell of Porous Metallic Fuel Containing a Sodium-Filled Pore and a Gas-Filled Pore (Only One Pore is Shown for Clarity).

P_{c1}, P_{c2} = fractions of the area of the x-z face of the unit cell that are occupied respectively by the gas-filled pore and the sodium-filled pore,

K_{pt1}, K_{pt2} = apparent thermal conductivities of the pore tubes containing the gas-filled pore and the sodium-filled pore.

The apparent thermal conductivity of a pore tube can be evaluated from the formula for two thermal resistances in series in the y-direction: (1) the pore proper, and (2) the solid fuel contained in the pore tube. For the two pore tubes one obtains the following equations:

$$\frac{1}{K_{pt1}} = \frac{P_{L1}}{K_g} + \frac{1 - P_{L1}}{K_s} \tag{10.3-73}$$

$$\frac{1}{K_{pt2}} = \frac{P_{L2}}{K_{Na}} + \frac{1 - P_{L2}}{K_s} \tag{10.3-74}$$

where

P_{L1}, P_{L2} = fractions of the side of the unit cell in the y-direction that are occupied respectively by the gas-filled pore and the sodium-filled pore,

K_g = thermal conductivity of the fission gas contained in the gas-filled pore,

K_{Na} = thermal conductivity of the sodium contained in the sodium-filled pore.

Neglecting K_g in comparison to K_s and K_{Na}, and solving Eq. (10.3-73) for K_{pt1}, it is found that K_{pt1} is zero. Substituting this value of K_{pt1} and the value of K_{pt2} obtained from Eq. (10.3-74) into Eq. (10.3-72), the effective thermal conductivity of the pours fuel can be written

$$\frac{K}{K_s} = \frac{P_{c2} K_{Na}}{K_s P_{L2} + (1 - P_{L2}) K_{Na}} + 1 - P_{c1} - P_{c2} \tag{10.3-75}$$

The gas-filled volume porosity, P_g, and the sodium-filled volume porosity, P_{Na}, can be written as

$$P_g = \frac{P_{c1}}{P_{L1}} \tag{10.3-76}$$

$$P_{Na} = \frac{P_{c2}}{P_{L2}} \tag{10.3-77}$$

For the cubic gas-filled pore of side a_1 shown in Fig. (Figure 10.3.71),

$$P_{c1} = \frac{a_1^2}{L^2} \tag{10.3-78}$$

$$P_{L1} = \frac{a_1}{L} \tag{10.3-79}$$

Eq. (10.3-76), Eq. (10.3-78) and Eq. (10.3-79) give the following relations of P_{c1} and P_{L1} to the gas-filled volume porosity P_g.

$$P_{c1} = P_g^{2/3} \tag{10.3-80}$$

$$P_{L1} = P_g^{1/3} \tag{10.3-81}$$

Similarly, P_{c2} and P_{L2} can be related to the sodium-filled volume porosity, P_{Na}, as follows:

$$P_{c2} = P_{Na}^{2/3} \tag{10.3-82}$$

$$P_{L2} = P_{Na}^{1/3} \tag{10.3-83}$$

Substituting Eq. (10.3-80), Eq. (10.3-82) and Eq. (10.3-83) into Eq. (10.3-75), the effective thermal conductivity of the porous fuel can be written as

$$\frac{K}{K_s} = 1 - P_g^{2/3} + \frac{P_{Na}^{2/3}}{\left(\frac{K_s}{K_{Na}}\right) P_{Na}^{1/3} + \left(1 - P_{Na}^{1/3}\right)} - P_{Na}^{2/3} \tag{10.3-84}$$

Eq. (10.3-84) gives the effect of sodium- and gas-filled porosities on fully dense fuel thermal conductivity. It is noted that Eq. (10.3-84) satisfies two limiting conditions: (1) in the absence of any logged sodium in the fuel, the ratio K/K_s equals the known reduction factor of $1 - P_g^{2/3}$ [10-26], and (2) if K_{Na} is set equal to K_s in Eq. (10.3-84), the last two terms on the right-hand side cancel to make K/K_s again equal to $1 - P_g^{2/3}$ which is expected when the sodium and the fully dense fuel behave alike and gas-filled porosity is the only nonconductive porosity in the material.

10.3.2.7 Uncertainties of U-Pu-Zr Fuel Thermal Conductivity

The uncertainties of U-Pu-Zr fuel thermal properties are important in the design and safety analyses of the Integral Fast Reactor (IFR). The purpose of the present section is to set the uncertainties of U-Pu-Zr fuel thermal conductivity that are used in the SAS4A/SASSYS-1 codes.

The uncertainties of U-Pu-Zr fuel thermal conductivity depend on burnup, and therefore the discussion of the subject is divided into two parts: (1) unirradiated U-Pu-Zr fuel, and (2) irradiated U-Pu-Zr fuel. As it will be seen below, the subject of uncertainties of U-Pu-Zr fuel thermal conductivity is a difficult one because of the limited data, and technical judgment has been used in the IFR Metallic Fuels Handbook [10-8, 10-9] to estimate uncertainties in a number of cases.

10.3.2.7.1 Unirradiated U-Pu-Zr Fuel

For evaluating composition- and temperature-dependent U-Pu-Zr fuel thermal conductivity, the IFR Handbook [10-8, 10-9] data have been incorporated into the SAS4A/SASSYS-1 codes. This evaluation uses a method of regionwise interpolation among quadratic fits to 9 basic unirradiated alloy/component data and Handbook U-Zr binary correlation [10-8]. The quadratic fits of the basic alloy data are done in the temperature variable. Table 10.3.7 lists the accuracy of the data for these 9 basic alloys/components, the maximum errors in the quadratic fits to these data, and the accuracy of the binary correlation. It should be pointed out that two of the required database alloys/components are taken from sources other than the Handbook, and a warning message is printed by the codes then these data are actually used in any calculation. The accuracy of data given in Table 10.3.7 is taken to represent 2 standard deviations (95% confidence limits) because accuracy implies that almost all the data lies within the accuracy band [10-31]. Assuming the accuracy to represent 1 standard deviation would imply that only about two-thirds of the data lies inside the accuracy band and about one-third of the data lie outside, and under this assumption almost all the data would not lie within the accuracy band. The maximum error in quadratic fits given in Table 10.3.7 is also taken to represent 2 standard deviations because all the data actually lies within the maximum error band above and below the fitted quadratic equation in temperature. Assuming the maximum error to represent 3 standard deviations is also justified but this is not done here. Statistically combining the accuracy of data and the error in quadratic fit for each database alloy/component (i.e., square-root of the sum of squares of percent accuracy and percent error), one obtains the combined uncertainty given in the last column of Table 10.3.7. The Pu metal has the largest combined 2-sigma uncertainty of 21.9% but most of this uncertainty arises from the error in the quadratic fit. If the 19.9% maximum error in Pu metal quadratic fit were assumed to represent 3 standard deviations, the combined 1-sigma uncertainty would become $\left[(19.9/3)^2 + (9.2/2)^2\right]^{1/2} = 8.1\%$. Thus Table 10.3.7 shows that the combined 1-sigman uncertainty can be taken as 10% for unirradiated U-Pu-Zr fuels. This also agrees with the conclusion of Ref. [10-21].

Table 10.3.7: Accuracy of Unirradiated 100% Dense U-Pu-Zr Alloy/Component Thermal Conductivity Data Used in the Method of Regionwise Interpolation

No.	Alloy/Component	Source of Data	Accuracy of Data	Max. Error in Quadratic Fit	Combined 2-sigma Uncertainty, %
1	U metal	Handbook	±20% [a]	0.72%	20.0
2	U-1.5 Zr	Handbook	±20% [a] ±5% [Ref 10-18]	0.41%	20.0
3	U-20 Zr	Handbook	±20% [a]	1.71%	20.1
4	U-40 Zr	Handbook	±20% [a]	3.13%	20.2
5	U-14.7Pu-9.7Ar	Handbook	±20% [a]	1.18%	20.0
6	U-18.4Pu-11.5Zr	Handbook	±20% [a]	2.57%	20.2
7	U-10Pu	Ref. [10-21]	Not reported	2.34%	–
8	Pu metal	Ref. [10-23]	±9.2% [b]	19.9%	21.9
9	U-16.2Pu-6.2Zr	Handbook	±20% [a]	2.74%	20.2
10	Handbook Binary Correlation	Handbook	±20% [a]	[c]	20.0

[a]Assumed in the Metallic Fuels Handbook [10-8].

[b]Based on two theoretical calculations of conductivity.

[c]The quoted accuracy includes fitting error also.

10.3.2.7.2 Irradiated U-Pu-Zr Fuel

No data is available for the thermal conductivity of irradiated U-Zr binary or U-Pu-Zr ternary fuel [10-8, 10-9]. But data are available for irradiated U-5Fs fuel from 3 experimental sources that are reviewed in the Handbook. It is assumed in the Handbook that U-Zr and U-Pu-Zr fuels behave similar to U-5Fs fuel during irradiation, and the same burnup correction factor may be used for all these fuels. The minimum value (just before porosity interlinkage) for irradiated-to-unirradiated thermal conductivity ratio, $K/K_{\rm o}$, recommended in the Handbook for U-Zr binary and U-Pu-Zr ternary fuels is 0.5 ± 0.1. This recommendation is based on rounding off the mean value of this ratio ($K/K_{\rm o}$ just before porosity interlinkage) and its standard deviation, 0.475 ± 0.111, reported by Betten [10-29] in one of the 3 experimental sources. This implies a standard deviation of 23.4%. The porosity interlinkage occurs in the 1-2 at. % burnup range [10-8, 10-9, 10-29], and it is assumed to occur at 2 at. % burnup in the SAS4A/SASSYS-1 codes, following a recent recommendation by Billone [10-21] used in Mark V fuel safety calculations. The standard deviation is assumed to vary linearly with burnup between unirradiated fuel and porosity interlinkage, i.e., zero to 2 at. %.

The long-time value (after porosity interlinkage, sodium logging and fuel hot pressing) for irradiated-to-unirradiated thermal conductivity ratio, $K/K_{\rm o}$, recommended in the Handbook [10-9] for U-Zr and U-Pu-Zr

fuels is 0.7 ± 0.1. This implies a standard deviation of 14.3%. The plots of K/K_{o} versus burnup given in Betten's experimental work [10-29] show that this long-time value is practically attained at about 5 at. % burnup. This value has also been recommended for Mark V fuel safety calculations [10-21]. The value of K/K_{o} and its standard deviation remain constant above this burnup as shown by the plots given in Betten's experimental work [10-29]. The standard deviation is assumed to vary linearly with burnup between porosity interlinkage and the long-time value attainment, i.e., 2 at. % to 5 at. %.

The conclusions made above about 1-sigma uncertainty can be summarized as follows:

$$1-\sigma \text{ uncertainties} = \begin{cases} \text{Varies linearly from 10\% to 23.4\% as burnup varies from 0 to 2 at. \% ,} \\ \text{Varies linearly from 23.4\% to 14.3\% as burnup varies from 2 to 5 at. \% ,} \\ \text{14.3\% for burnup greater than 5 at. \% .} \end{cases} \tag{10.3-85}$$

At present the following values of 1-sigma uncertainty have been incorporated in the SAS4A/SASSYS-1 codes. These values are rounded off and slightly higher than those concluded above. These values are also identical to the uncertainties recommended for Mark V fuel safety calculations [10-21].

$$1-\sigma \text{ uncertainties} = \begin{cases} \text{Varies linearly from 10\% to 25\% as burnup varies from 0 to 2 at. \% ,} \\ \text{Varies linearly from 25\% to 15\% as burnup varies from 2 to 5 at. \% ,} \\ \text{15\% for burnup greater than 5 at. \% .} \end{cases} \tag{10.3-86}$$

10.3.3 Correlations of the Report ANL/RAS 85-19

This section describes the composition and temperature-dependent correlations for thermal properties of U-Pu-Zr alloy fuel which are used by the earlier SSCOMP model [10-4] and other SAS4A/SASSYS-1 modules. These correlations were developed prior to the issuing of the Metallic Fuels Handbook [10-9] in order to provide an early SAS4A/SASSYS-1 capability to compute U-Pu-Zr fuel behavior. These correlations for evaluating fuel thermal properties are specified by setting the input parameter IRHOK =1 and the input parameter IFUELM =1 in a channel.

10.3.3.1 Theoretical Density of U-Pu-Zr Fuel at 298K

The theoretical density of U-Pu-Zr fuel at a room temperature of 298K is estimated from the theoretical densities of U, Pu and Zr by assuming that there is no volume change due to alloying of the constituents. This assumption is discussed in more detail below. The theoretical densities of natural (isotopic composition) U at 298K. 239Pu at 294K, and Zr at 293K are found from their crystal structures to be 19.04 g/cm^3, 19.816 g/cm^3 and 6.52 g/cm^3, respectively [10-14]. The minor corrections required in the densities of Pu and Zr due to the small differences between the quoted temperature and the room temperature 298K are made by using the following coefficients of linear expansion given in the same reference.

$$\alpha\left(^{239}\mathrm{Pu}\right) = (46.85 + 0.0559T_{\mathrm{c}}) \times 10^{6}, \quad -186 < T_{\mathrm{c}} < 100 \tag{10.3-87}$$

$$\alpha\left(\mathrm{Zr}\right) = (5.768 + 0.006154T_{\mathrm{c}}) \times 10^{-6}, \quad 0 < T_{\mathrm{c}} < 600 \tag{10.3-88}$$

where

α = coefficient of linear thermal expansion per °C,

T_{c} = temperature on the Centigrade scale.

The theoretical density of ^{239}Pu at 298K after the temperature correction turns out to be 19.805 g/cm^3, and that of Zr at 298K is about 6.51 g/cm^3 because its temperature correction is negligible for the 5K difference. The

theoretical densities of U and Pu can be further corrected for the difference between the actual isotopic average atomic weight and the quoted atomic weight, by multiplying by the ratio of these atomic weights.

$$\rho_{\text{th, u}}\,(298\text{K}) = 19040\frac{a_{\text{u}}}{238.07} \tag{10.3-89}$$

$$\rho_{\text{th, p}}\,(298\text{K}) = 19805\frac{a_{\text{p}}}{239.13} \tag{10.3-90}$$

where

$\rho_{\text{th,u}}\rho_{\text{th,p}}$ = theoretical densities of U and Pu at 298K, kg/m^3,

a_{u} = uranium atomic weight averaged over its actual isotopic composition in fuel,

a_{p} = plutonium atomic weight averaged over its actual isotopic composition in the fuel.

The denominators 238.07 and 239.13 of Eq. (10.3-89) and Eq. (10.3-90) are the atomic weights of natural uranium and ^{239}Pu.

The volume of one kilogram of U-Pu-Zr alloy at 298K can be found from Eq. (10.3-89) and Eq. (10.3-90) by summing the volumes of its constituents.

$$V_{\text{s}} = \frac{238.07W_{\text{u}}}{19040a_{\text{u}}} + \frac{239.13W_{\text{p}}}{19805a_{\text{p}}} + \frac{W_{\text{z}}}{6510}\ , \tag{10.3-91}$$

where

V_{a} = volume of one kilogram of U-Pu-Zr alloy at 298 K, m^3/kg,

$W_{\text{u}}, W_{\text{p}}, W_{\text{z}}$ = weight fractions of U, Pu and Zr in the alloy.

The theoretical density of U-Pu-Zr alloy at 298K in units of kg/m^3 is obtained from Eq. (10.3-91).

$$\rho_{\text{th}}\,(298\text{K}, W_{\text{p}}, W_{\text{z}}) = \frac{1000}{\left(\frac{238.07W_{\text{u}}}{19.04a_{\text{u}}} + \frac{239.13W_{\text{p}}}{19.805a_{\text{u}}} + \frac{W_{\text{z}}}{6.51}\right)} \tag{10.3-92}$$

where

$\rho_{\text{th}}\,(T, W_{\text{p}}, W_{\text{z}})$ = theoretical density of U-Pu-Zr alloy at temperature T°K, kg/m^3.

Table 10.3.8: Comparison of Measured and Estimated Theoretical Densities of U-Pu-Zr Alloy at 298K

Quantity	Batch R250	Batch R253Z
U, Pu and Zr weight ratios	73.4/13.8/12.2	75.9/14.8/9.4
U, Pu and Zr weight fractions	0.738/0.139/0.123	0.758/0.148/0.094
Isotopic composition of U, ^{235}U/^{238}U wt %	93.19/6.81	93.19/6.81
Isotopic composition of U, ^{239}Pu/^{240}Pu/^{242}Pu wt %	91.35/7.85/0.77/0.035	91.35/7.85/0.77/0.035
Average atomic weight of U	235.32	235.32
Average atomic weight of Pu	239.22	239.22
Estimated (using Eq. (10.3-92)) theoretical density of U-Pu-Zr fuel, kg/m^3	15360	16080
Measured density averaged over all the fuel slugs in the batch, kg/m^3	15280	15750

Table 10.3.8 shows a comparison of measured densities of two batches of injection cast U-Pu-Zr fuel slugs [10-3] with the theoretical densities estimated from Eq. (10.3-92). The comparison is satisfactory for these two fuel compositions although further comparison for alloys of other compositions, specifically the alloy compositions in

the annular zones formed at the end of steady-state irradiation of U-Pu-Zr fuel pins, is required to test the validity of the assumption of no volume change during alloying. The small difference may be due to measurement error, small voids left during the injection casting process or some volume change taking place during alloying.

10.3.3.2 Thermal Expansion of U-Pu-Zr Fuel

The U-Pu-Zr alloys show strong discontinuities in the coefficient of linear thermal expansion when the alloys are heated through the phase transformation temperatures [10-15]. There is some variation between the coefficients of linear expansion of alloys of different compositions but there is not enough data available to correlate this variation as a function of composition. This fuel properties option assumes the coefficient of linear expansion to be independent of composition, and uses the thermal expansion of U-15 wt % Pu-10 wt % Zr for all alloy compositions. The measured values of U-15 wt % Pu-10 wt % Zr alloy linear thermal expansion have been correlated as follows:

$$\frac{\Delta L\left(T\right)}{L_1} = \begin{cases} 1.76 \times 10^{-5}\left(T - 298\right), & T \leq 873 \text{ K} \\ 1.012 \times 10^{-2} + 1.05 \times 10^{-4}\left(T - 873\right), & 873 < T \leq 923 \text{ K} \\ 1.537 \times 10^{-2} + 2.04 \times 10^{-5}\left(T - 923\right), & T > 923 \text{ K} \end{cases} \tag{10.3-93}$$

where

$\frac{\Delta L(T)}{L_1}$ = linear thermal expansion from a fixed temperature 298K to any temperature T, expressed as a fraction of length L_1 at the temperature 298K.

Using Eq. (10.3-92) and Eq. (10.3-93), the following equation for the theoretical density of U-Pu-Zr alloy fuel at any temperature T in units of kg/m^3 is obtained:

$$\rho_{\text{th}}\left(T, W_{\text{p}}, W_{\text{z}}\right) = \frac{1000}{\left(\frac{238.07 W_{\text{u}}}{19.04 a_{\text{u}}} + \frac{239.13 W_{\text{p}}}{19.805 a_{\text{p}}} + \frac{W_{\text{z}}}{6.51}\right)} \frac{1}{\left[1 + \frac{\Delta L(T)}{L_1}\right]^3} \tag{10.3-94}$$

10.3.3.3 Enthalpy and Specific Heat of U-Pu-Zr Fuel

The composition-dependent enthalpy and specific heat of U-Pu-Zr alloy at any temperature up to is solidus temperature are estimated by mass-weighted averaging of the enthalpy and specific heat data of U, Pu and Zr metals, assuming that the heat of formation of the alloy is negligible. The heat of formation of the alloy is assumed negligible due to lack of data.

Each of the constituent metals of the U-Pu-Zr alloy undergoes certain crystallographic phase transformations in the solid state which require considerable heat addition and enthalpy increase at the transformation temperatures. The specific heat data of U, Pu, and Zr excluding these heats of solid-state transformations [10-10] are given in Tables *10.3-9* to *10.3-11*, and plotted in Figure 10.3.72 to Figure 10.3.74. In each plot, the value of specific heat between the highest solid-state transformation temperature and the melting point is shown extrapolated to a temperature of 2000K for curve-fitting purposes, although the fitted equations are used only up to a solidus temperature of the U-Pu-Zr alloy. Because the heat transfer routines of the SAS4A/SASSYS-1 codes were initially written for the mixed-oxide fuel without any heat of solid-heat transformation, the heats of solid-state transformations of U, Pu and Zr are required to be included in their specific heats by spreading the corresponding enthalpy jumps over a temperature range. This spreading is not all artificial because these solid-state transformations actually occur over a temperature range in the U-Pu-Zr alloy. Hence the equations fitted to the specific heat and enthalpy data of U, Pu and Zr metals are required (1) to be continuous (no sudden jumps at any temperature) for the purpose of accuracy and convergence of the numerical solution in the SAS4A/SASSYS-1 heat transfer routines, (2) to include the heats of solid-state transformations such that the fitted equations cross the enthalpy jumps close to their midpoints, and (3) to closely reproduce the enthalpy and specific heat data away from the solid-state transformations. When one polynomial could not be fitted over the whole temperature range,

the data for each metal was fitted in two parts: a polynomial over the lower temperature range covering most of the solid-state transformations, and an equation of the form a+b/T^n over the higher temperature range. The following equations for the specific heats (in units of J/kg-K) of U, Pu and Zr metals and the U-Pu-Zr alloy are obtained.

$$C_{\text{pu}}(T) = \begin{cases} 0.72516T - 1.4232 \times 10^{-3}T^2 + 1.0134 \times 10^{-6}T^3 , & T \leq 1000\ \text{K} \\ 160.9 + 154.5\left(\frac{1000}{T}\right)^8 , & T > 1000\ \text{K} \end{cases} \tag{10.3-95}$$

$$C_{\text{pp}}(T) = \begin{cases} -1202.3 + 9.0307T - 0.018232T^2 + 1.1616 \times 10^{-5}T^3 , & T \leq 753\ \text{K} \\ 147.0 + 72.65\left(\frac{753}{T}\right)^9 , & T > 753\ \text{K} \end{cases} \tag{10.3-96}$$

$$C_{\text{pz}}(T) = \begin{cases} -7.0215 + 1.5493T - 2.4845 \times 10^{-3}T^2 + 1.3587 \times 10^{-6}T^3 , & T \leq 1136\ \text{K} \\ 344.2 + 196.4\left(\frac{1136}{T}\right)^9 , & T > 1136\ \text{K} \end{cases} \tag{10.3-97}$$

$$C_{\text{p}}(T, W_{\text{p}}, W_{\text{z}}) = W_{\text{u}}C_{\text{pu}}(T) + W_{\text{p}}C_{\text{pp}}(T) + W_{\text{z}}C_{\text{pz}}(T) \tag{10.3-98}$$

where

$C_{\text{pu}}(T), C_{\text{pp}}(T), C_{\text{pz}}(T)$ = specific heats of U,Pu and Zr metals at temperature T, J/kg-K,

$C_{\text{p}}(T, W_{\text{p}}, W_{\text{z}})$ = composition-dependent specific heat of the U-Pu-Zr alloy, J/kg-K,

$W_{\text{u}}, W_{\text{p}}, W_{\text{z}}$ = weight fractions of U,Pu and Zr in the U-Pu-Zr alloy.

Eq. (10.3-95) to Eq. (10.3-97) for the specific heats of U, Pu and Zr metals are compared with the data of Tables *10.3-9* to *10.3-11* in Figure 10.3.72 to Figure 10.3.74. The lower and the higher temperature-range-fitted equations join without any discontinuity at the range interface temperature. The enthalpies obtained by integrating Eq. (10.3-95) to Eq. (10.3-97) are compared with the measured data of Tables *10.3-9* to 10.3-11 in Figure 10.3.75 to Figure 10.3.77. The enthalpy obtained from the higher temperature-range-fitted equation merges asymptotically with the data. A comparison of the enthalpy obtained from Eq. (10.3-98) for alloys of several compositions (e.g., those formed in the annular zones) is required to test the validity of this technique.

10.3.3.4 Thermal Conductivity of U-Pu-Zr Fuel

The thermal conductivity of U-Pu-Zr alloy (up to its solidus temperature) as a function of temperature and composition has been obtained by evaluating the effect of alloying plutonium on the thermal conductivity of U-Zr binary alloy. The thermal conductivity of U-Zr binary alloy is known for the full range of composition [7-14] and is shown in Figure 10.3.78. Alloying usually reduces the thermal conductivity substantially. The reduction in the U-Zr binary thermal conductivity due to alloying Pu with it is evaluated using the thermal conductivity data available for three U-Pu-Zr ternary alloys prepared in casting numbers D-44, D-54 and D-50 [10-19, 10-20] and given in Table 10.3.12.

Figure 10.3.78 shows that the thermal conductivity of U-Zr binary alloy decreased linearly with Zr atom fraction in the range of zero to 0.5. Therefore, quadratic functions are least-square fitted to the pure uranium thermal conductivity data and to the U-50 atom % zr binary alloy data of Figure 10.3.78, and then the thermal conductivity of any U-Zr binary alloy in the composition range of zero to 50 at. % Zr is linearly interpolated between these two quadratics. The resulting equations are as follows:

$$K_{\text{u}} = 0.22173 + 1.8562 \times 10^{-4}T_{\text{c}} + 1.3278 \times 10^{-7}T_{\text{c}}^2 \tag{10.3-99}$$

$$K_{\text{u50}} = 0.074766 + 1.6738 \times 10^{-4}T_{\text{c}} + 1.1599 \times 10^{-7}T_{\text{c}}^2 \tag{10.3-100}$$

$$K_{\text{uz}} = K_{\text{u}} + \frac{2A_{\text{z}}}{A_{\text{u}} + A_{\text{z}}}(K_{\text{u50}} - K_{\text{u}}) \tag{10.3-101}$$

Table 10.3.9: Enthalpy and Specific Heat Data of Uranium Metal [10-10]

Temperature, K	Specific Heat	Enthalpy Above	
	cal/mol-K	**J/kg-K**	**298.15K, cal/mol**
298.15	6.61	116.2	0.0
400.0	7.10	124.8	698
500.0	7.65	134.5	1434
600.0	8.31	146.1	2231
700.0	9.08	159.7	3099
800.0	9.99	175.7	4050
900.0	11.02	193.8	5103
941.0 (α)	11.47	201.7	5563
941.0 (β)	10.26	180.4	6230
1000.0	10.26	180.4	6836
1048.0 (β)	10.26	180.4	7327
1048.0 (γ)	9.15	160.9	8464
1100.0	9.15	160.9	8939
1200.0	9.15	160.9	9854
1300.0	9.15	160.9	10769
1400.0	9.15	160.9	11684
1500.0	9.15[a]	160.9[a]	12599[a]
2000.0	9.15[a]	160.9[a]	17174[a]

[a] The constant value of the γ-U-phase specific heat has been extrapolated to temperatures up to 2000K for curve-fitting purposes.

Table 10.3.10: Enthalpy and Specific Heat Data of Plutonium Metal [10-10]

Temperature, K	Specific Heat	Enthalpy Above	
	cal/mol-K	**J/kg-K**	**298.15K, cal/mol**
298.15	7.64	133.7	0.0
395.0 (α)	8.20	143.5	772
395.0 (β)	8.00	140.0	1612
400.0	8.03	140.6	2277
480.0 (β)	8.60	150.5	2417
480.0 (γ)	8.30	145.3	2585
500.0	8.53	149.3	3378
588.0 (γ)	9.50	166.3	3508
600.0 (δ)	9.00	157.5	3616
700.0	9.00	157.5	4516
730.0 (δ)	9.00	157.5	4786
730.0 (δ')	9.00	157.5	4806
753.0 (δ')	9.00	157.5	5013
753.0 (ε)	8.40	147.0	5453
800.0	8.40	147.0	5848
900.0	8.40	147.0[a]	6688
1000.0	8.40[a]	147.0[a]	7528[a]
2000.0	8.40[a]	147.0[a]	15928[a]

[a] The constant value of the ε-U-phase specific heat has been extrapolated to temperatures up to 2000K for

curve-fitting purposes.

Table 10.3.11: Enthalpy and Specific Heat Data for Zirconium Metal [10-10]

Temperature, K	Specific Heat	Enthalpy Above	
	cal/mol-K	**J/kg-K**	**298.15K, cal/mol**
298.15	6.06	278.1	0.0
400.0	6.54	300.1	646
500.0	6.78	311.1	1313
600.0	7.01	321.7	2003
700.0	7.23	331.8	2714
800.0	7.45	341.9	3449
900.0	7.68	352.4	4205
1000.0	7.90	362.5	4984
1100.0	8.13	373.1	5785
1136.0 (α)	8.20	376.3	6079
1136.0 (β)	7.50	344.2	7020
1200.0	7.50	344.2	7500
1400.0	7.50	344.2	9000
1600.0	7.50	344.2	10500
1800.00	7.50	344.2	12000
2000.0	7.50	344.2	13500

where

T_c = temperature of the alloy, °C,

A_u, A_p, A_z = atom fractions of U, Pu and Zr in the U-Pu-Zr ternary alloy. For a U-Zr binary alloy, A_p is zero.

K_u = thermal conductivity of pure uranium, W/cm-°C,

K_{u50} = thermal conductivity of U-50 atom % Zr binary alloy, W/cm°C,

K_{uz} = thermal conductivity of any U-Zr binary alloy in the composition range of zero to 50 atom % Zr, W/cm-°C.

Eq. (10.3-99) and Eq. (10.3-100) reproduce the thermal conductivity data shown in Figure 10.3.78 for pure uranium and U-50 atom % Zr binary alloy with a maximum error of 1.5%. Eq. (10.3-101) for the thermal conductivity of U-Zr binary alloys has been compared in Table 10.3.13 with the data [10-18] for U-20 wt. % Zr (U-39.5 atom & Zr) binary alloy and the maximum error is about 5%. Eq. (10.3-101) is so written that it determines the thermal conductivity of the U-Zr binary alloy to which Pu can be added to obtain the desired U-Pu-Zr ternary alloy with U, Pu and Zr atom fractions equal to A_u, A_p and A_z. The addition of Pu to the U-Zr binary alloy decreases the thermal conductivity. Because of a lack of the ternary alloy thermal conductivity data at widely spaced Pu atom fractions, only a linear dependence on Pu atom fraction is assumed in this evaluation.

$$K_o\left(T_c, A_p, A_z\right) = K_{uz} - CA_p \tag{10.3-102}$$

where

$K_o\left(T_c, A_p, A_z\right)$ = thermal conductivity at temperature T_c °C of the solid (i.e., nonporous) U-Pu-Zr ternary alloy having U, Pu and Zr atom fractions equal to A_u, A_p and A_z, W/cm-°C.

In Eq. (10.3-102), the coefficient C is independent of A_p and models the effect of alloying Pu with the U-Zr binary.

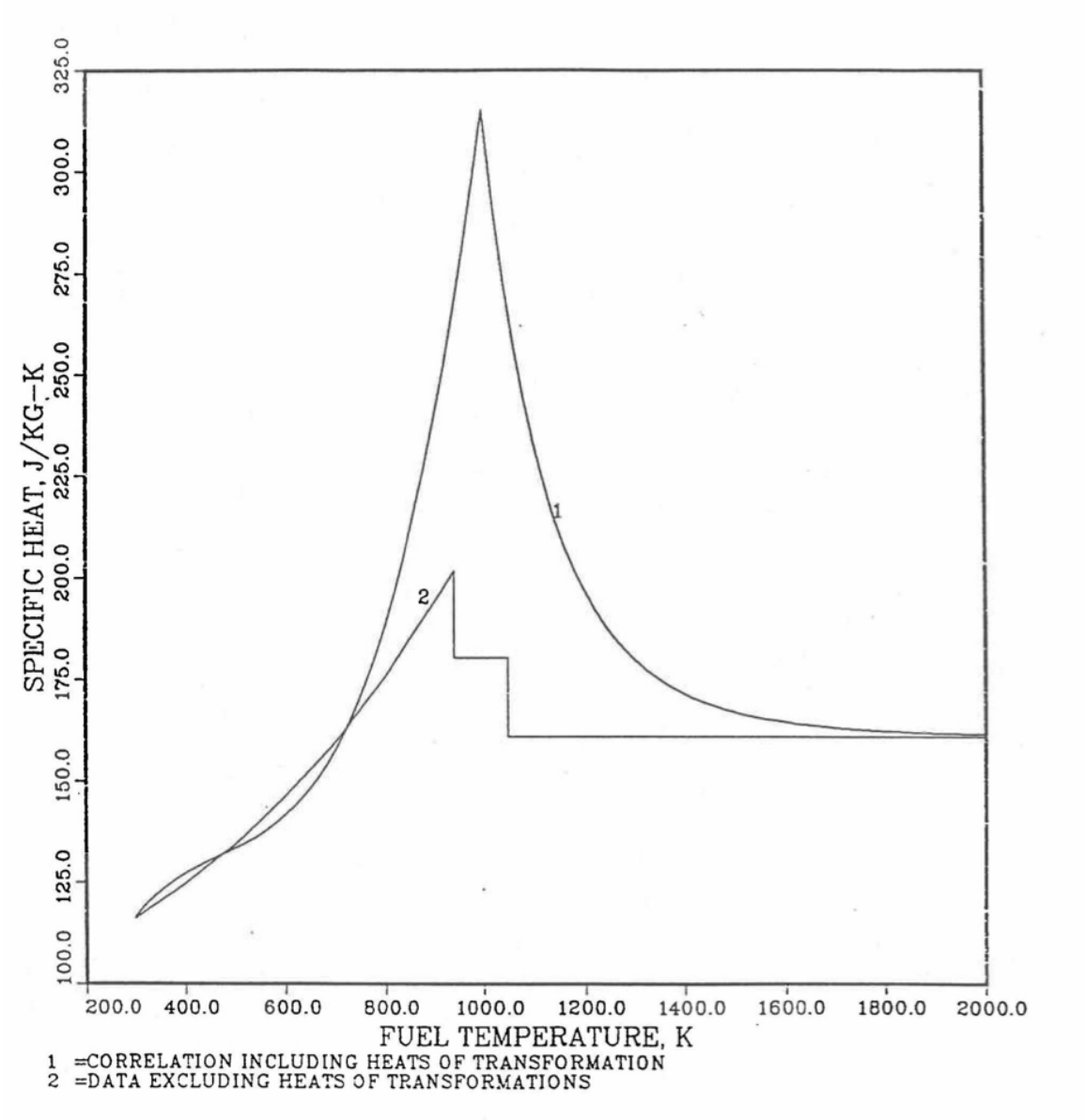


Figure 10.3.72: Uranium Specific Heat Correlation Including the Heats of SolidState Transformations.

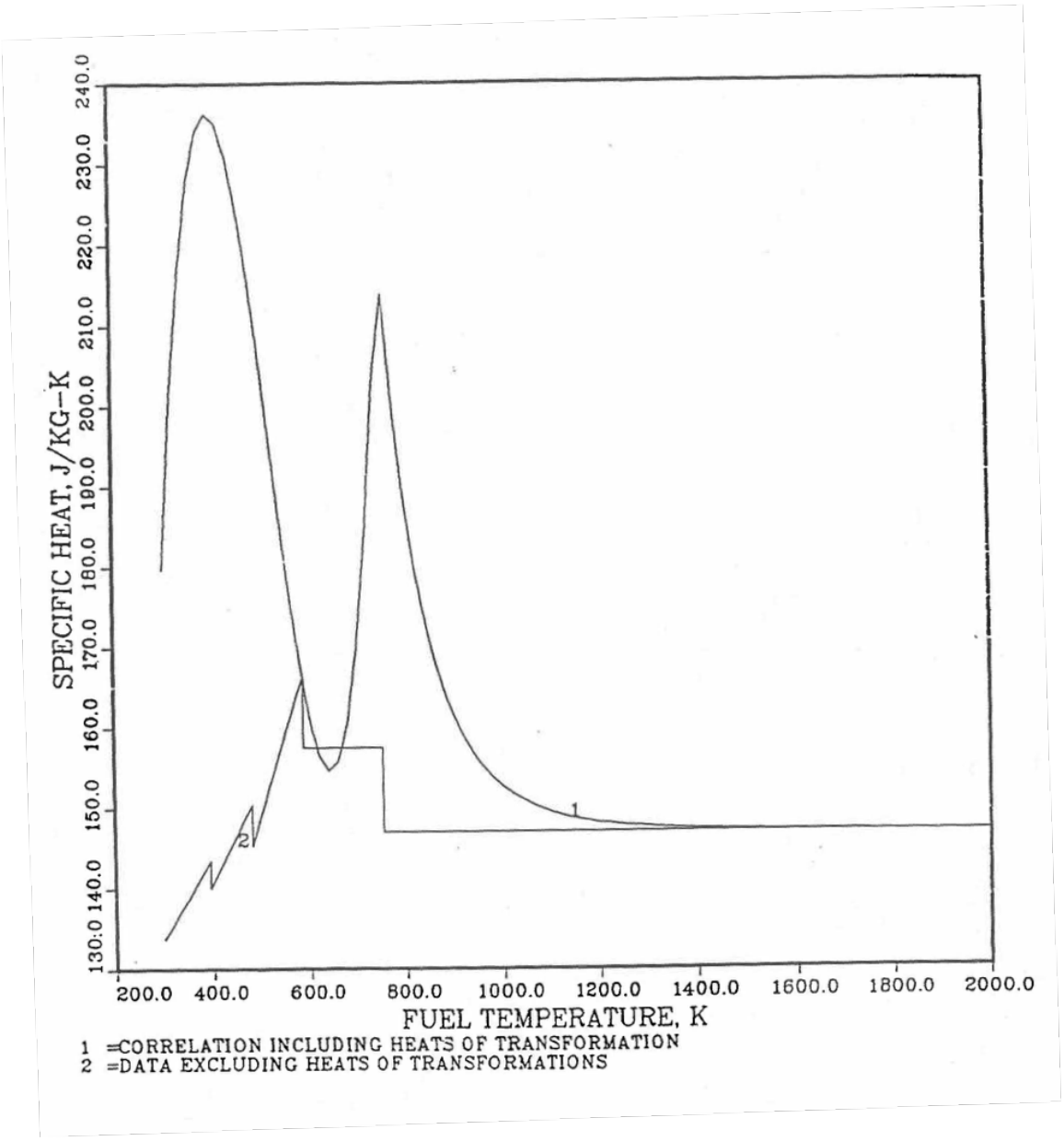


Figure 10.3.73: Plutonium Specific Heat Correlation Including the Heats of SolidState Temperature.

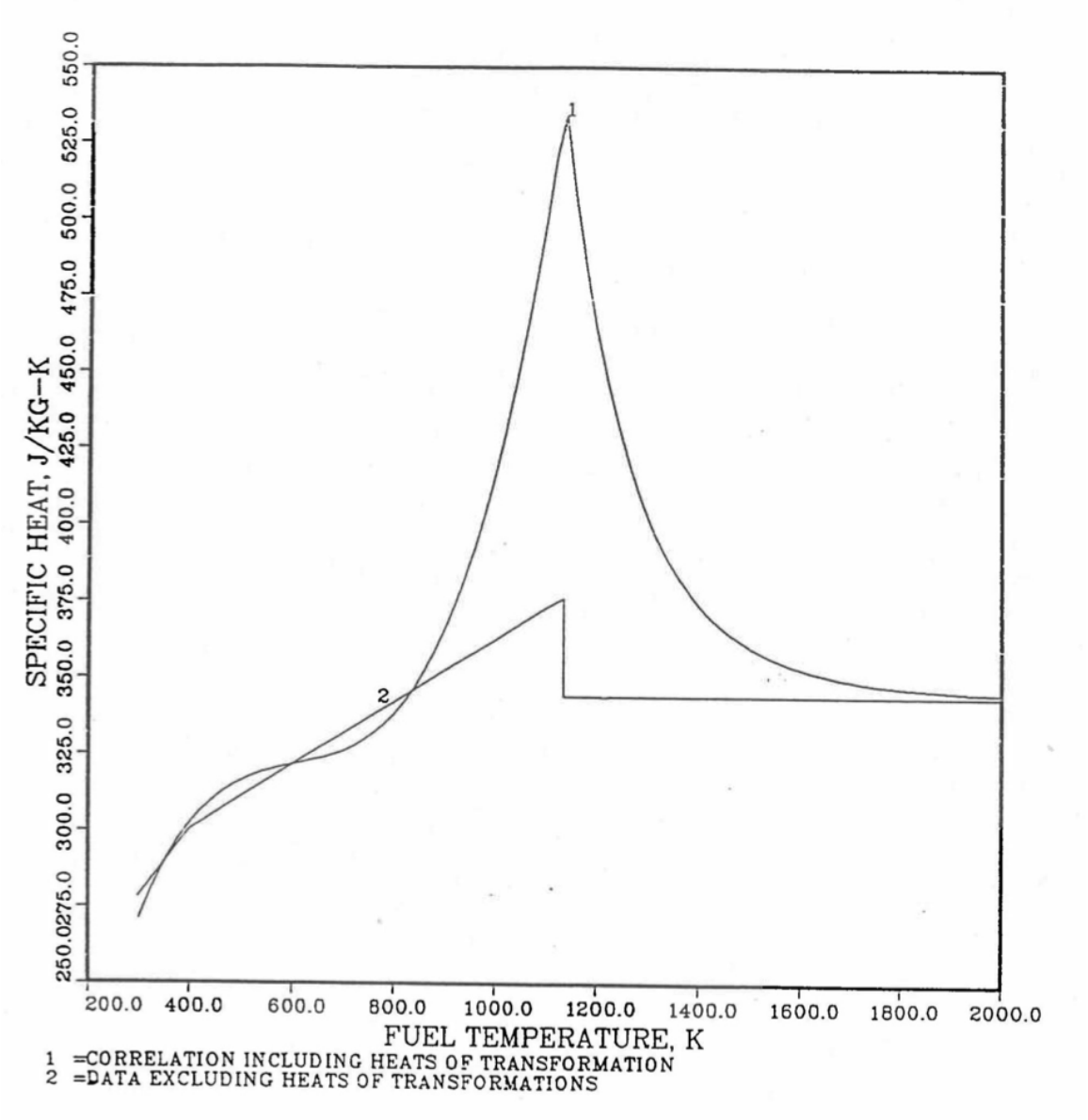


Figure 10.3.74: Zirconium Specific Heat Correlation Including the Heats of SolidState Transformations.

Table 10.3.12: Thermal Conductivity Data for Three U-Pu-Zr Alloys Prepared in Casting Numbers D-44, D-54 and D50

Temperature, °C	Thermal Conductivity, W/cm-°C Casting D-44[a]	Casting D-54[a]	Casting D-50[a]
100	0.109	0.100	0.092
200	0.142	0.132	0.117
300	0.177	0.164	0.144
400	0.210	0.196	0.171
500	0.241	0.225	0.197
600	0.264	0.253	0.219
700	0.280	0.278	0.234
800	0.293	0.301	0.248
900	0.301	0.323	0.260
1100[b]	0.322	0.368	0.286
1300[b]	0.343	0.413	0.312
1500[b]	0.364	0.458	0.338
1700[b]	0.385	0.503	0.364
1900[b]	0.406	0.548	0.390
2000[b]	0.417	0.571	0.403

[a] The composition of castings:

D-44 is U-16.2 wt & Pu-6.2 wt % Zr (U-14.7 at. % Pu-14.7 at.% Zr

D-54 is U-14.7 wt & Pu-9.7 wt % Zr (U-12.7 at. % Pu-21.9 at.% Zr

D-50 is U-18.4 wt & Pu-11.5 wt % Zr (U-15.5 at. % Pu-25.3 at.% Zr

[b] The data in Ref. [10-19, 10-20] have been extrapolated linearly to temperatures up to 2000°C for curve fitting purposes.

Table 10.3.13: Comparison of Thermal Conductivity Evaluated Using Eq. (10.3-101) With the Data [10-18] for U-20Zr Binary Alloy

Temperature, °C	Thermal Conductivity, W/cm-°C		
	Data	**Equation**	**% Error**
20	0.11	0.109	-0.8
100	0.13	0.124	-4.6
200	0.15	0.145	-3.5
300	0.17	0.168	-1.3
400	0.20	0.193	-3.4
500	0.22	0.221	0.5
600	0.25	0.251	0.6
700	0.28	0.284	1.5
800	0.31	0.319	2.9
900	0.34	0.357	4.9

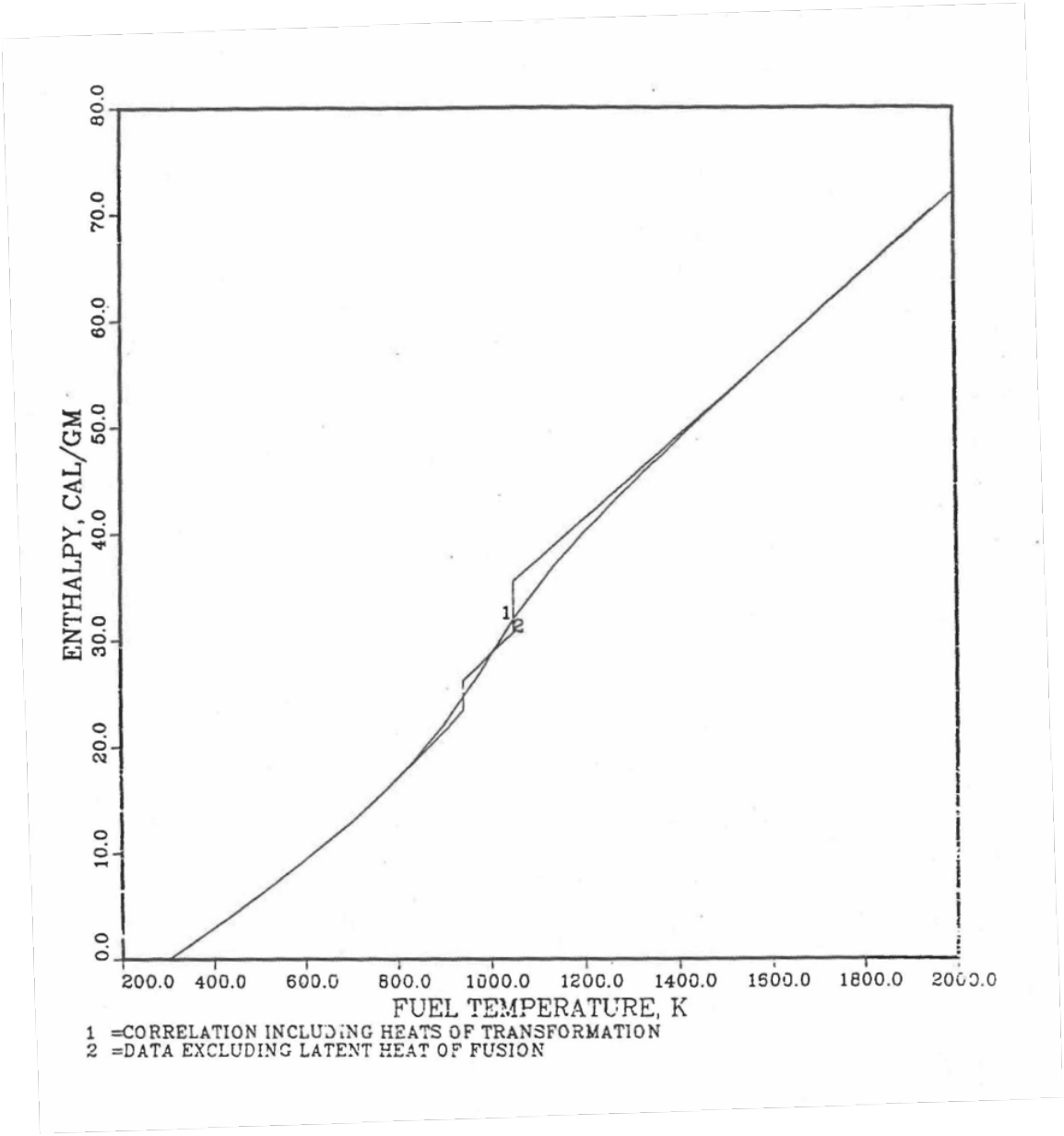


Figure 10.3.75: Uranium Enthalpy Obtained by Integrating its Specific Heat Correlation.

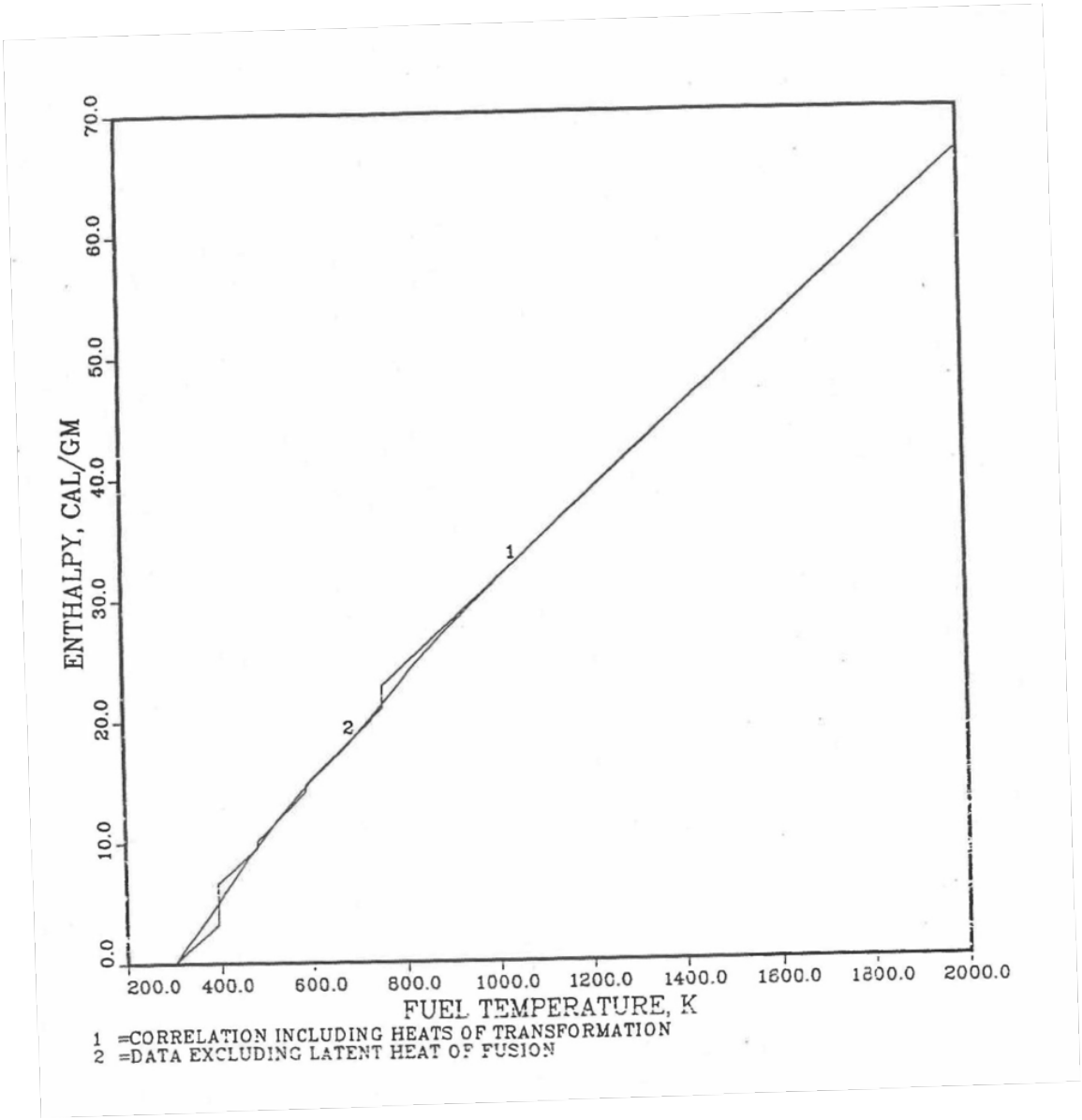


Figure 10.3.76: Plutonium Enthalpy Obtained by Integrating its Specific Heat Correlation.

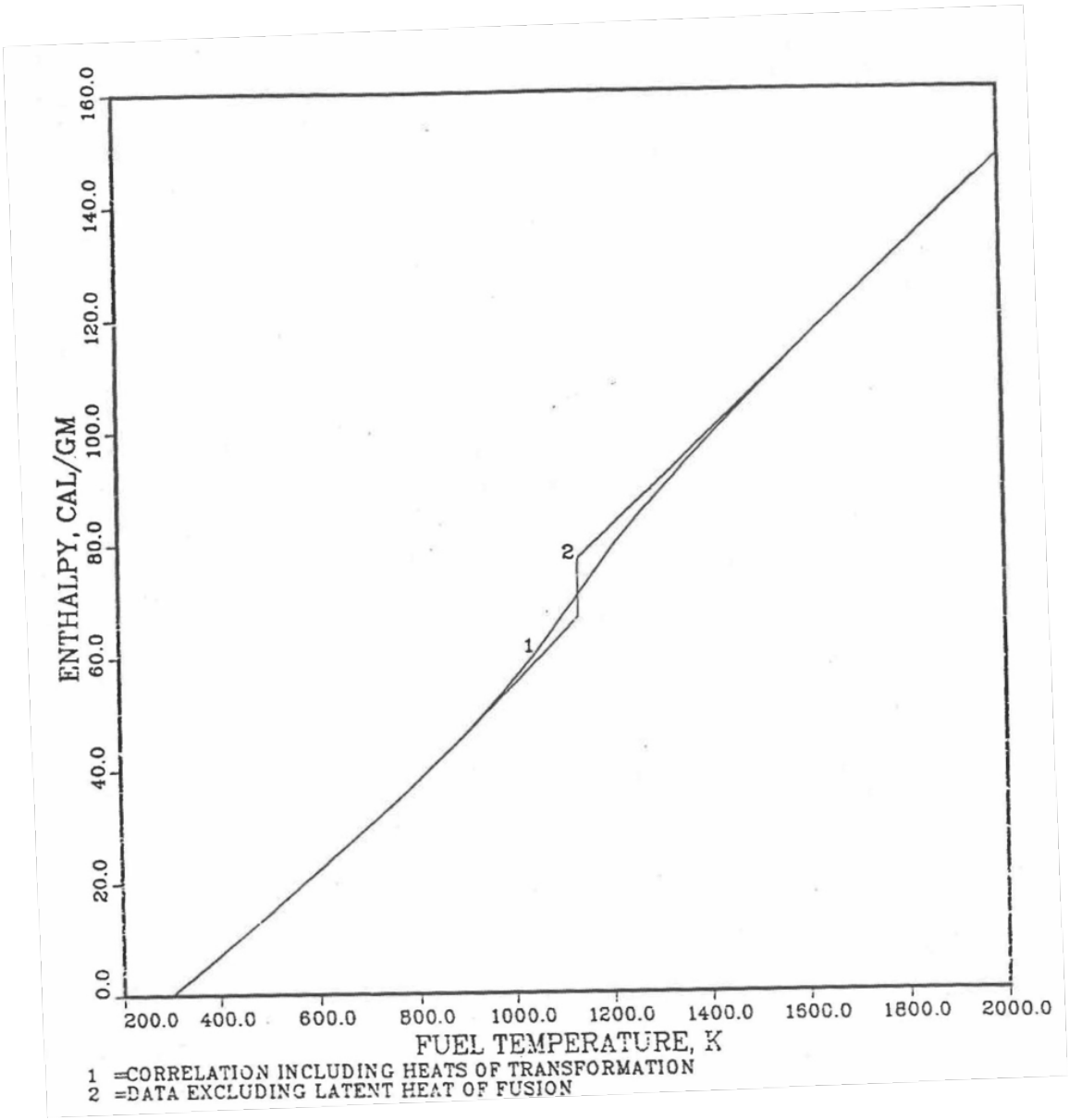


Figure 10.3.77: Zirconium Enthalpy Obtained by Integrating its Specific Heat Correlation.

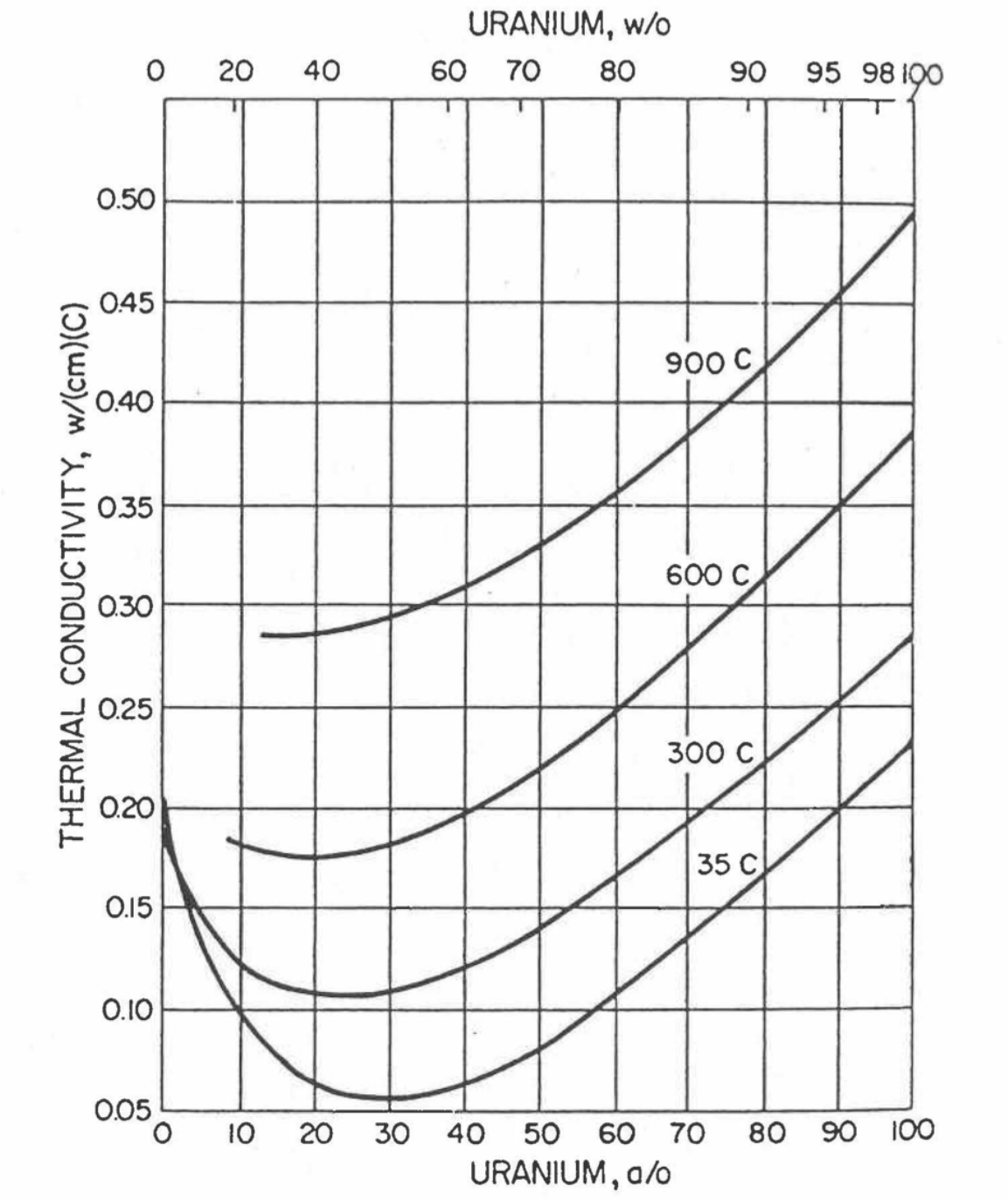


Figure 10.3.78: Thermal Conductivity of the Uranium-Zirconium Binary Alloy for the Full Range of Composition.

Table 10.3.14: Temperature Variation of the Plutonium Alloying Coefficient C Evaluated from the Thermal Conductivity Data for Three U-Pu-Zr Alloys Prepared in Casting Numbers D-44, D-54 and D-50

Temperature, °C	**Thermal Conductivity, W/cm-°C** U-Pu-Zr Casting	U-Zr Binary With U to Zr Atom Ratio Same as Casting	Coefficient C
Casting Number D-44, U-14.7 at.% Pu-14.7 at.% Zr Alloy			
100	0.109	0.190	0.554
200	0.142	0.212	0.477
300	0.177	0.236	0.404
400	0.210	0.263	0.362
500	0.241	0.292	0.351
600	0.264	0.324	0.412
700	0.280	0.359	0.537
800	0.293	0.396	0.701
900	0.301	0.435	0.916
Casting Number D-54, U-12.7 at.% Pu-21.9 at.% Zr Alloy			
100	0.100	0.167	0.529
200	0.132	0.188	0.445
300	0.164	0.212	0.381
400	0.196	0.238	0.336
500	0.225	0.267	0.335
600	0.253	0.299	0.361
700	0.278	0.332	0.431
800	0.301	0.369	0.536
900	0.323	0.408	0.668
Casting Number D-50, U-15.5 at.% Pu-25.3 at.% Zr Alloy			
100	0.092	0.152	0.390
200	0.117	0.173	0.365
300	0.144	0.197	0.344
400	0.171	0.223	0.337
500	0.197	0.252	0.354
600	0.219	0.283	0.412
700	0.234	0.316	0.531
800	0.248	0.352	0.673
900	0.260	0.390	0.843

The coefficient C has been evaluated (see table 10.3-14) from the thermal conductivity data given in Table 10.3.12 for three U-Pu-Zr alloys prepared in casting numbers D-44, D-54 and D-50 [10-19, 10-20], and Figure 10.3.79 shows a plot of these values of the coefficient C versus temperature. The scatter in Figure 10.3.79 indicates that the coefficient C may have a weak dependence on Zr atom fraction in addition to the main dependence on temperature. The following two equations have been developed for evaluating the coefficient C: (1) a second degree polynomial of temperature only (ignoring the weak dependence on Zr atom fraction) least-

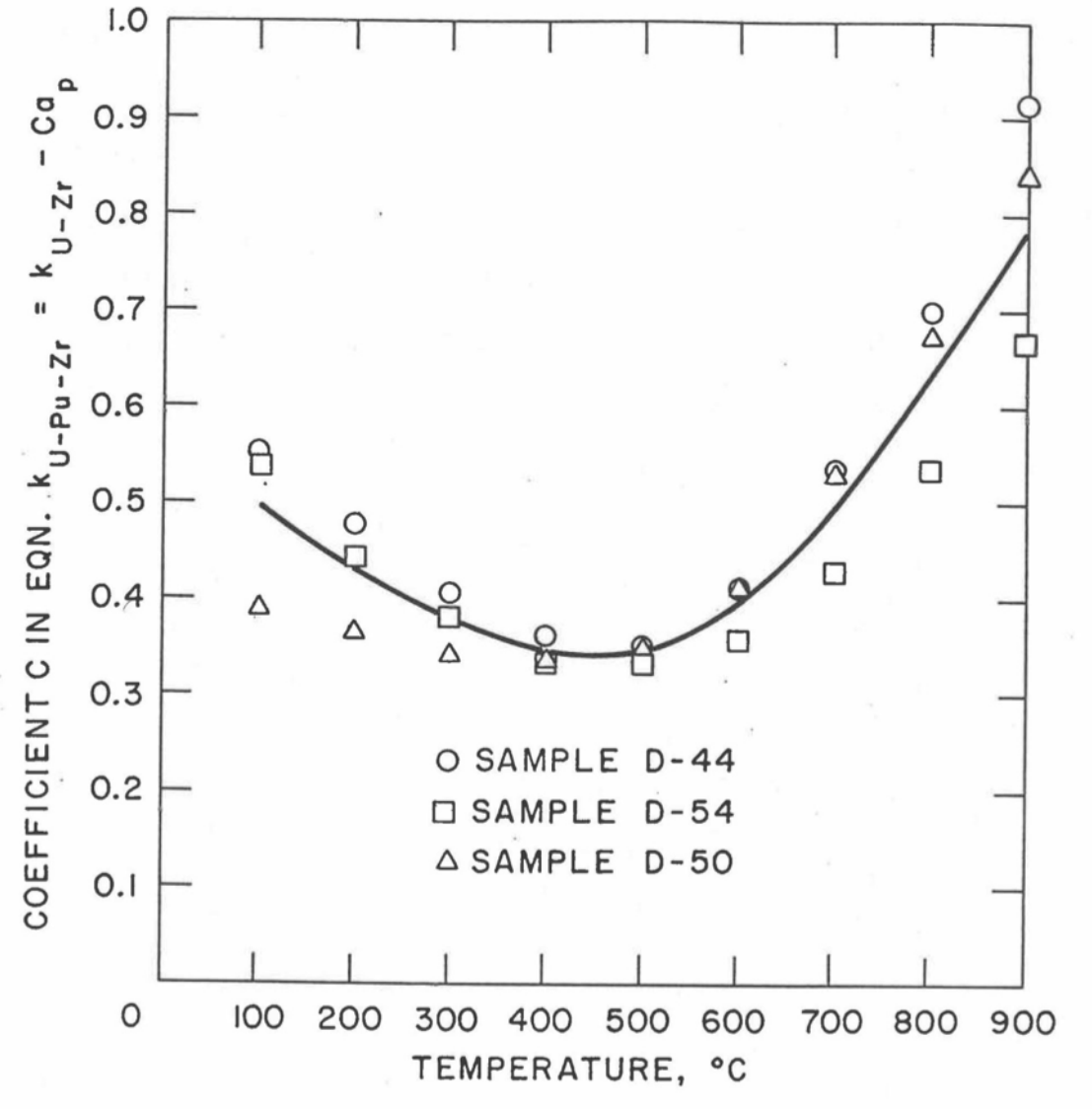


Figure 10.3.79: Plot of Coefficient C vs. Temperature Obtained from Thermal Conductivity Data for U-Pu-Zr Samples D-44, D-54 and D-50.

squared fitted to the data given in Table 10.3.12 up to 900°C using A_p^2 as the weighting factor, and (2) a second degree polynomial of temperature with its coefficients linearly depending on Zr atom fraction least-square fitted to all the data given in Table 10.3.12 up to 2000°C using $[K_o(T_c, A_p, A_z)]^{-2}$ as the weighting factor. The A_p^2 weighting factor was used to reduce the errors in the first fitting equation at higher Pu atom fractions and make them nearly equal to the errors at lower Pu atom fractions. The $1/K_p^2$ weighting factor was used in the second case for the purpose of minimizing the mean fractional error in thermal conductivity. The fitted equations are given below with their expected ranges of validity.

$$C = 0.63477 - 1.4980 \times 10^{-3} T_c + 1.8878 \times 10^{-6} T_c^2, \quad 100 \le T_c \le 1200, \;\; 0.0 \le A_p \le 0.16, \;\; 0.0 \le A_z \le 0.50 \tag{10.3-103}$$

$$C = (0.89079 - 1.6757 A_z) - T_c \left(1.7437 \times 10^{-3} - 4.0008 \times 10^{-3} A_z\right) + T_c^2 \left(1.9608 \times 10^{-6} - 3.1043 \times 10^{-6} A_z\right), \quad 100 \le T_c \le 1500, \;\; 0.0 \le A_p \le 0.20, \;\; 0.0 \le A_z \le 0.50 \tag{10.3-104}$$

Figure 10.3.80 to Figure 10.3.82 compare the U-Pu-Zr alloy thermal conductivity obtained from Eq. (10.3-102) and Eq. (10.3-104) with the data for the three U-Pu-Zr alloys prepared in casting numbers D-44, D-54 and D-50 [10-19, 19-20]. The correlation compares well with all the base data with a root mean square error of 5% and a maximum error of about 10%. Eq. (10.3-104) for evaluating the coefficient C covers a wider range of composition and temperature than Eq. (10.3-103) and is preferred. Further comparison for alloys of other compositions (i.e., those formed in the annular zones) and at temperatures higher than 900°C is required to test the validity of the assumptions used in this correlation.

The IFR Handbook correlation given by Eq. (10.7-12) reproduces all the base data of Table 10.3.12 over the temperature range 100-900°C with a maximum error of about 20% compared to the maximum error of about 10% in the correlation Eq. (10.3-102) and Eq. (10.3-104).

10.3.3.5 Porosity Correction of Fuel Thermal Conductivity

The correlation for the thermal conductivity of unirradiated 100%-dense U-Pu-Zr alloy fuel is described in Section 10.3.3.4 for the fuel properties option specified by setting the input parameter IRHOK=1 and the input parameter IFUELM=1 in a channel. The thermal conductivity also depends strongly on the fission gas-filled and logged sodium-filled porosities that form in the fuel during normal steady-state irradiation. Two methods to correct the thermal conductivity of 100%-dense U-Pu-Zr alloy fuel for fission gas-filled and logged sodium-filled porosities have been described in Section 10.3.2.6: (i) an empirical method based on experimental data for U-5Fs alloy fuel, and (ii) a theoretical method that explicitly accounts for the gas- and sodium-filled porosities. Only the theoretical method is available for irradiation-effects correction in this fuel properties option as shown by the flow diagrams of subroutines FK and KFUEL in Figure 10.3.3 and Figure 10.3.5.

10.3.4 Properties of Some Specific Fuels

Before the implementation in SAS4A/SASSYS-1 codes of composition- and temperature- dependent fuel thermal properties by the method of regionwise interpolation [10-31] of the Metallic Fuels Handbook [10-8] data, thermal properties of two specific fuels, i.e., Mark-V (U-20Pu-10Zr) ternary and U-10Zr binary alloy fuels, were incorporated into the codes by Briggs [10-32] to perform EBR-II Mark-V fuel safety calculations. These thermal properties are in the form of temperature-dependent correlations that were developed by Billone [10-12, 10-21, 10-22] based on data reported in the Metallic Fuels Handbook [10-8]. For the purpose of reproducing these safety calculations if required in future, these thermal property correlations have been retained in the codes and are used by setting the input parameter IRHOK-1 and the input parameter IFUELM-2 in a channel. In this fuel properties option, the specific heat and theoretical density of Mark-V ternary and U-10Zr binary fuels (specified by setting the input parameter IMETAL = 2 or 3 respectively) are evaluated using the following correlations:

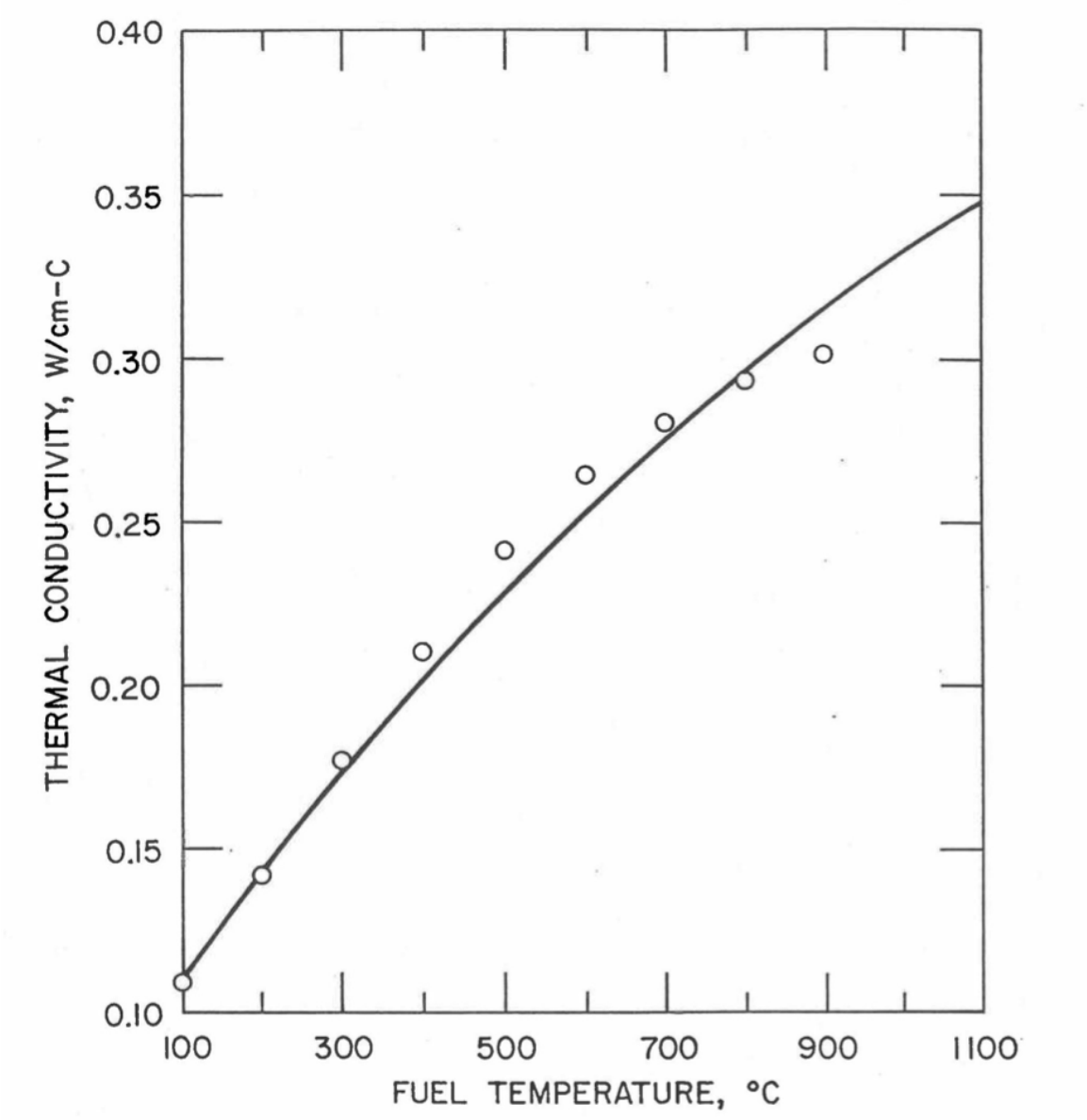


Figure 10.3.80: Comparison of the Thermal Conductivity Correlation Eq. (10.3-102) and Eq. (10.3-104) with the Data for the U-Pu-Zr Sample D-44.

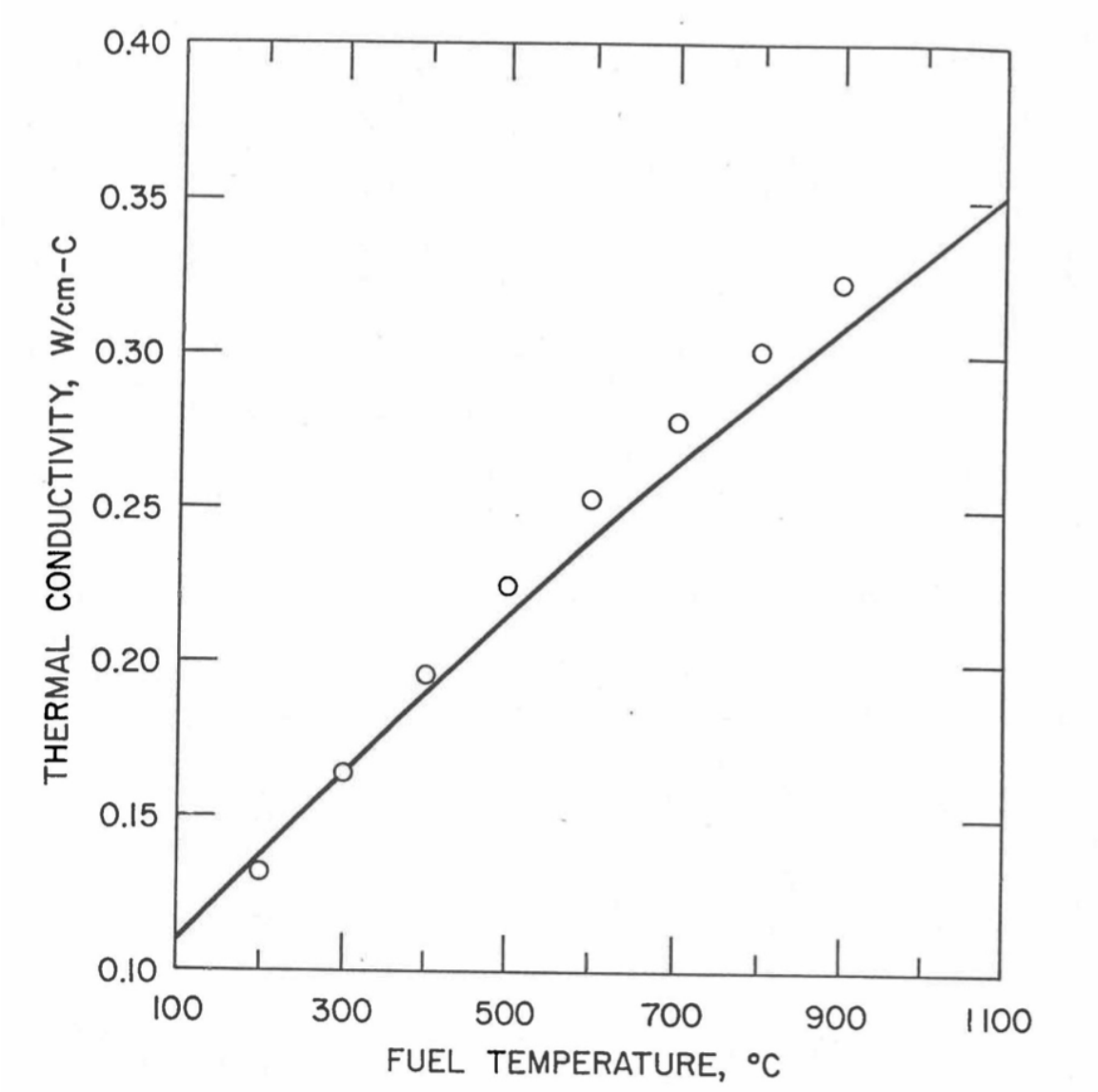


Figure 10.3.81: Comparison of the Thermal Conductivity Correlation Eq. (10.3-102) and Eq. (10.3-104) with the Data for the U-Pu-Zr Sample D-54.

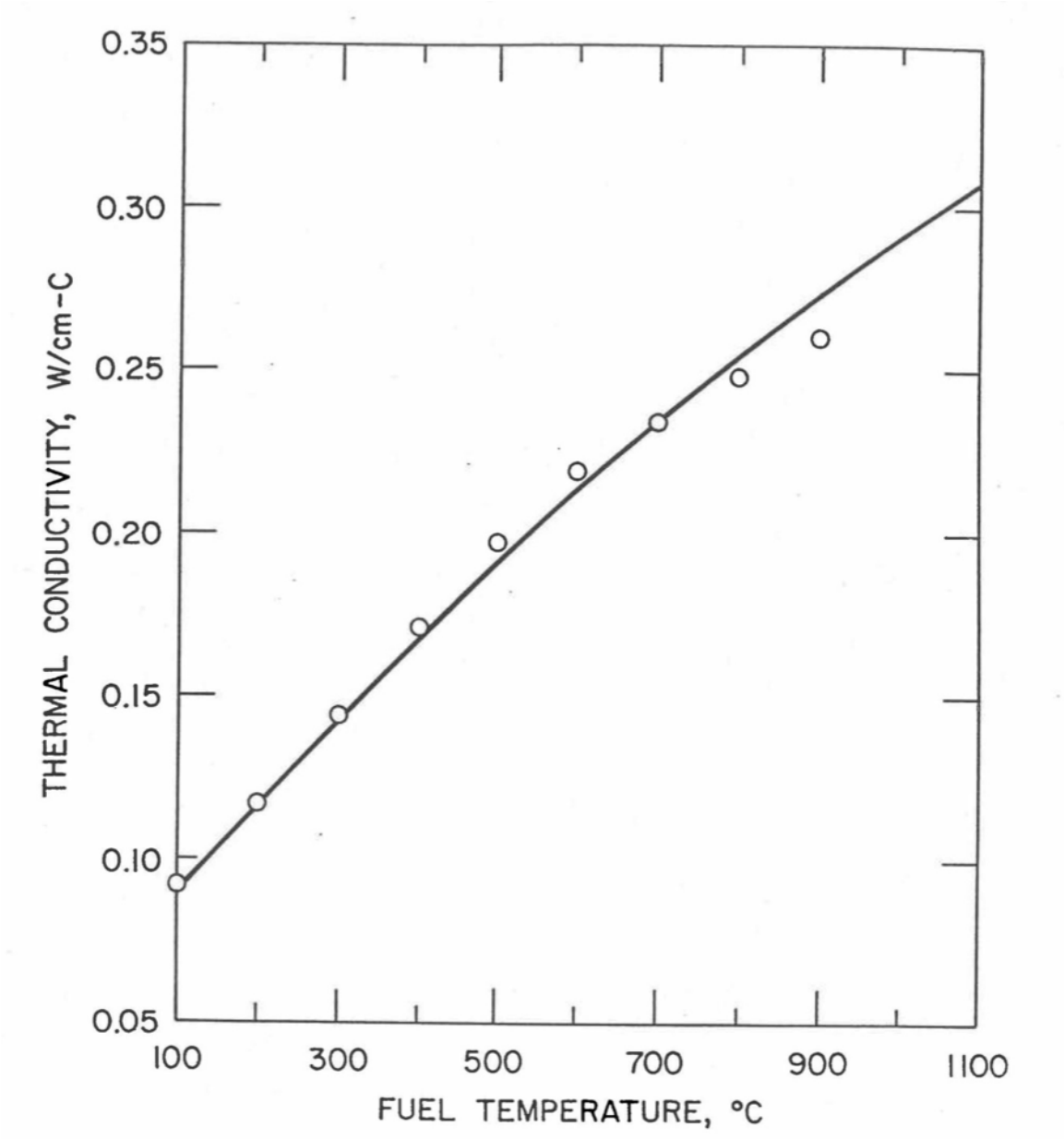


Figure 10.3.82: Comparison of the Thermal Conductivity Correlation Eq. (10.3-102) and Eq. (10.3-104) with the Data for the U-Pu-Zr Sample D-50.

Mark-V (U-20Pu-10Zr) Fuel (IMETAL=2)

$$C_{\mathrm{p}}(T) = \begin{cases} 94.62 + 0.1301T, T < 873 \text{ K} & \\ 792.6, & 873 \leq T < 923 \text{ K} \\ 42.82 + 0.1276T, & 923 \leq T < 1379 \text{ K} \\ 440.1, & 1379 \leq T < 1588 \text{ K} \\ 217.2, & T \geq 1588 \text{ K} \end{cases} \tag{10.3-105}$$

$$\rho_{\mathrm{th}}(T) = 15730 \left[1 + \frac{\Delta L(T)}{L_{\mathrm{o}}}\right]^{-3} \tag{10.3-106}$$

$$\frac{\Delta L(T)}{L_{\mathrm{o}}} = \begin{cases} 1.73 \times 10^{-5}(T - 293), & T \leq 859 \text{ K} \\ \frac{\Delta L(859)}{L_{\mathrm{o}}} + \frac{5.8 \times 10^{-3}}{69}(1.655 - 6.55W_{\mathrm{z}})(T - 859), & 859 \leq T \leq 928 \text{ K} \\ \frac{\Delta L(928)}{L_{\mathrm{o}}} + 2.29 \times 10^{-5}\left(1 + 0.567W_{\mathrm{z}} - 7.42W_{\mathrm{z}}^2\right)(T - 928), & T \geq 928 \text{ K} \end{cases} \tag{10.3-107}$$

U-10Zr binary Fuel (IMETAL=3)

$$C_{\mathrm{p}}(T) = \begin{cases} 6.625 + 0.3066T + \frac{4.58 \times 10^6}{T^2}, & T < 1000 \text{ K} \\ 180.1, & 1000 \leq T < 1506 \text{ K} \\ 580.7, & 1506 \leq T < 1669 \text{ K} \\ 221.9, & T \geq 1669 \text{ K} \end{cases} \tag{10.3-108}$$

$$\rho_{\mathrm{th}}(T) = 16.02 \times 10^3 \left(1.0122 - 4.629 \times 10^{-5}T + 2.438 \times 10^{-8}T^2 - 2.805 \times 10^{-11}T^3\right) \tag{10.3-109}$$

where

C_{p} = specific heat, J/kg-K

T = fuel temperature, K

ρ_{th} = theoretical density, kg/m^3

$\frac{\Delta L(T)}{L_{\mathrm{o}}}$ = linear thermal expansion from 293K to temperature T, expressed as a fraction of length L_{o} at the temperature 293K.

$W_{\mathrm{z}} = 0.1$, the zirconium weight fraction in the fuel.

It is important to note that the specific heats of Mark-V and U-10Zr fuels over the melting range, obtained from Eq. (10.3-105) and Eq. (10.3-108), are the actual values, and are not consistent with the method of Section 3.3.5 for adjusting fuel temperature (computed without accounting for heat of fusion during the main heat transfer calculation) to account for heat of fusion of any radial mesh interval that is going through the melting range. To be consistent with the method (see subroutine TSHTN5), the fuel specific heat used here must be the value for solid fuel just below solidus or the value for molten fuel just above liquidus.

The unirradiated 100%-dense fuel thermal conductivity, for both Mark-V and U-10Zr fuels, is evaluated using Eq. (10.7-13) from the atom fractions of Pu and Zr in the fuel. These atom fractions are evaluated in the SAS4A/SASSYS-1 codes, using the following relations in this fuel properties option.

$$A_{\mathrm{p}} = \frac{W_{\mathrm{p}}}{(1.0101 - 0.0101W_{\mathrm{p}} + 1.611W_{\mathrm{z}})} \tag{10.3-110}$$

$$A_{\mathrm{z}} = 1.627\frac{W_{\mathrm{z}}}{(0.6272 - 0.006297W_{\mathrm{p}} + W_{\mathrm{z}})} \tag{10.3-111}$$

where

A_p, A_z = atom fractions of Pu and Zr in the fuel,

W_p, W_z = weight fractions of Pu and Zr in the fuel.

As discussed in Section 10.7.2, this thermal conductivity correlation should not be used for any other alloy composition besides Mark-V and U-10Zr fuels because it is ill-behaved, and so recommended by Billone [10-22]. For these two fuels, this correlation has been found to be correct. The correction of the 100% dense fuel thermal conductivity for the effects of steady-state irradiation is done using the empirical method described in Section 10.3.2.6. For the purpose of reproducing EBR-II Mark-V fuel safety calculations if required in future (which is the main purpose of providing this fuel properties option), the theoretical method of correcting for irradiation effects is not available in this option. Finally, to account for the uncertainties in the U-Pu-Zr thermal conductivity data forming the basis of this correlation (i.e., Eq. (10.7-13)), the thermal conductivity obtained after correcting for irradiation effects is divided by an input uncertainty factor FUNKFU.

10.3.5 Computer Implementation

The U-Pu-Zr alloy composition in each computational cell of the fuel is at present specified both axially and radially by input data determined from the examination fuel pins with similar irradiation histories. Input data are handled by annular fuel zones (central, middle and outer composition zones) but the codes internally have arrays by radial node to store U-Pu-Zr fuel composition and composition-dependent quantities. The choice of input data by fuel zone is based on the fact that analysts communicate commonly in terms of zones, and that this simplifies input data preparation. The use of storage by radial node is based on the fact that this reduces logical IF statements from coding, especially in molten material motion models. This should simplify the codes and improve the running time. The addition of composition-dependent fuel thermal properties in the input data is an option in the codes, and the user should be able to run no-zone formation cases without preparing the input data for multiple fuel zones. This section describes the functions of all the fuel thermal property subroutines, the related input data, and internal arrays.

10.3.5.1 List of Fuel Thermal Property Subroutines

The following is a list of all the fuel thermal property subroutines with a brief description of their functions:

Name	Descriptions
CFUEL	Computes specific heat of mixed-oxide, U-Fs and U-Pu-Zr alloy fuels for a range of fuel radial nodes in an axial segment, with specified temperatures and U-Pu-Zr compositions.
CFUEL1	Computes U-Pu-Zr alloy fuel specific heat as a function of composition and temperature by regionswise interpolation of Metallic Fuels Handbook enthalpy data.
FK	Computes thermal conductivity of mixed-oxide, U-Fs and U-Pu-Zr alloy fuels for a single fuel radial node in an axial segment, with specified temperature, porosity and U-Pu-Zr composition. In the case of metal fuels, it also accounts for the effect of sodium logging.
HFUEL1	Computes U-Pu-Zr alloy fuel enthalpy as a function of composition and temperature by regionwise interpolation of Metallic Fuels Handbook enthalpy data.
KFUEL	Computes thermal conductivity of mixed oxide, U-Fs and U-Pu-Zr alloy fuels for a range of fuel radial nodes in an axial segment, with specified temperatures, porosity and U-Pu-Zr compositions. In the case of metal fuels, it also accounts for the effect of sodium logging.
KFUEL1	Computes thermal conductivity of unirradiated 100%-dense U-Pu-Zr alloy fuel as a function of composition and temperature, by the method of regionwise interpolation of Metallic Fuels Handbook thermal conductivity data.
PELTON	Computes solidus and liquidus temperatures of U-Pu-Zr alloy fuel as a function of composition, by the method of chemical thermodynamic equilibrium between conjugate solid and liquid solutions.
PRECAL	Ppre-calculates, before steady-stat calculation, a number of parameters stored for fast evaluation of U-Pu-Zr specific heat, theoretical density and thermal conductivity by fuel type.
RHOF	Computes theoretical density of mixed-oxide, U-Fs and U-Pu-Zr alloy fuels for a specified temperature and U-Pu-Zr composition.
RHOFM	Computes theoretical density of U-Pu-Zr fuel as function of composition and temperature, and also returns the fractional linear thermal expansion over the input temperature range.
THEXP1	Computes fractional linear thermal expansion of U-Pu-Zr alloy fuel as a function of composition and temperature, by the method of regionwise interpolation of Metallic Fuels Handbook data.
TRTEMP	Evaluates the first and last solid-state transition temperatures of U-Pu-Zr alloy fuel as a function of composition.

10.3.5.2 Input Data Related to Fuel Thermal Properties

Table 10.3.15 gives the SAS4A/SASSYS-1 input variables required to use (i) U-Pu-Zr fuel thermal properties obtained by regionwise interpolation of IFR Handbook data, and (ii) multiple radial fuel zones option of the codes. The first column of the table gives the location of the variable in a complete input deck of the codes. The second column gives the Fortran variable name used in the codes and the standard input description (see Section 2.7). Suggested values of some of these input variables are shown in column 3.

The suggested value, 5.0K, of the input variable DTFUEL (i.e., the maximum fuel temperature change per heat transfer time step) is based on a preliminary study of errors in computed fuel temperatures due to specific heat discontinuity at solid-state transitions. A utility program was developed and used to find the maximum temperature difference between two SAS4A/SASSYS-1 calculations with a main time step of 0.5 sec., differing only in the values used for the input variable DTFUEL, 50.0K and 1.0K in the two calculations. Errors of up to 16K, or up to 32% of the temperature rise in the main time step were found. A value of 5.0K for the input

variable DTFUEL should reduce the maximum error in fuel temperatures printed at main time steps to (16 x 5.0/50.0) 1.6K.

By setting the input variable IRHOK=1 and the input variable IFUELM=0 in a channel, one uses the method regionwise interpolation of IFR Handbook data to obtain U-Pu-Zr fuel thermal properties as a function of composition and temperature, and this is the recommended option among the built-in fuel thermal property options. By setting the input variable IFUELC=1, one uses the multiple radial fuel zones option instead of the single radial fuel zone, and then assigns fuel types to central, middle and outer fuel zones using the input array MFTZN instead of the input variables IFUELV and IFUELB.

Table 10.3.15 gives the internal arrays in SAS4A/SASSYS-1 common blocks that are required for implementing IFR Handbook U-Pu-Zr fuel properties data and multiple radial fuel zones option in the codes. Arrays FUFEMS and FUNIMS, for iron and nickel mass distribution in the fuel, have been included in the common block for future modeling work on radial migration of cladding constituents into the fuel. Arrays ACPF, AKFU and ADFU for pre-migration coefficients in equations for fuel specific heat, thermal conductivity and theoretical density, have a first dimension of 8 to represent the 8 fuel types allowed in the codes. The second dimension of each of these arrays represents the number of coefficients that can be stored for each fuel type. The 7 coefficients in the specific heat equation, stored as array elements ACPR(*,1), ACPR(*,2),…ACPF(*,7), are defined in Figure 10.3.18. Only 3 array elements AKFU(*,1), AKFU(*,2) and AKFU(*,3) are currently used to store the 3 pre-calculated coefficient of the quadratic for 100% dense fuel thermal conductivity, given by Eq. (10.3-61). Only one array element ADFU(*,1) is currently used to store pre-calculated theoretical density without fission products at 293K, for each fuel type, given by Eq. (10.3-32).

Table 10.3.15: Input Data Required to Use IFR Handbook U-Pu-Zr Fuel Properties Data and Multiple Radial Fuel Zones Option

Input Location	Fortran Variable	Suggested Value	Definition/Comments
Block 1, INPCOM			
92	KHDBK	0	controls printing of messages if any extrapolation of Handbook data is used.
93	KFIRR	0	Option of method for correcting thermal conductivity for irradiation effect.
94	KDENBU		controls whether theoretical density is corrected for the presence of fission products or not.
Block 11, OPCIN			
10	DTFUEL	5.0	This value of DTFUEL is suggested to keep below 1.6K, the errors in computed fuel temperatures due to specific heat discontinuity at solid-state transitions.
Block 13, PMATCM			
786-793	TFSOL		Solidus temperature by fuel type. If Handbook U-Pu-Zr fuel properties option is used, then a zero entry implies that the code will internally evaluate solidus temperature.
794-801	TFLIQ		Liquidus temperature by fuel type. If Handbook U-Pu-Zr tuel properties option is used, then a zero entry implies that the code will internally evaluate liquidus temperature.

continues on next page

Table 10.3.15 – continued from previous page

Input Location	Fortran Variable	Suggested Value	Definition/Comments
802-809	UFMELT		Heat of fusion by fuel type. If the IFR Handbook U-Pu-Zr fuel properties option is used, then the input is replaced by a value internally calculated by the code.
1073-1080	PRSTY		Total (i.e., fission gas-filled and logged sodium-filled) porosity by fuel type.
1300-1315	PUZRTP		Pu and Zr weight fractions by fuel type.
1316-1323	RHOZN	-1.0	U-Pu-Zr fuel theoretical density at the reference temperature TR, by fuel type. If RHOZN = 0.0 for a fuel type, then the implication is that this fuel type does not use the Handbook U-Pu-Zr fuel properties of input option IFUELM=0.
1324-1331	XLOGNA		Porosity fraction logged by bond sodium by fuel type
1332	XSIGMC	0.0	To account for uncertainties in Handbook data for U-Pu-Zr fuel specific heat, this input is the number of standard deviations to be added to the best estimate. It is not currently used.
1333	XSIGMK	0.0	To account for uncertainties in Handbook data for U-Pu-Zr fuel thermal conductivity, this input is the number of standard deviations to be added to the best estimate.
1334	XSIGMD	0.0	To account for uncertainties in Handbook data for U-Pu-Zr fuel theoretical density, this input is the number of standard deviations to be added to the best estimate. It is not currently used.
Block 51, INPCHN			
3	IRHOK	1	Selection parameter for the form (i.e., input tables or built-in correlations) of fuel thermal properties.
15	IFUELV		Fuel type or properties table number to be used for core fuel, in each channel.
16	IFEULB		Fuel type or properties table number to be used for blanket fuel, in each channel.
189	IMETAL	2	Input parameter to indicate whether the fuel is a mixed oxide or a metallic alloy of some kind, i.e., U-Fs, U-Pu-Zr or U-Zr binary.
191	IFUELO	0	Option to indicate whether annular fuel zone formation data (i.e., zonal compositions and radii) at the end of steady-state irradiation is obtained form user input or computed form the SSCOMP physical model.
192	IFUELM	0	Option to indicate the source of fuel thermal properties data. A zero entry implies the regionwise interpolation of IFR Handbook data.
193	IFUELC	1	An input flag to specify whether there is a single radial fuel zone, or there are multiple radial fuel zones.

continues on next page

Table 10.3.15 – continued from previous page

Input Location	Fortran Variable	Suggested Value	Definition/Comments
366-389	IZNC		Outermost radial mesh interval of central zone in each axial segment. This input determines central zone boundary only if the input RIZNC, central zone outer radius, is set to zero.
390-413	IZNM		Outermost radial mesh interval of central zone in each axial segment. This input determines central zone boundary only if the input RIZNM, central zone outer radius, is set to zero.
414-485	MFTZN		This input array is used (instead of IFUELV and IFUELB) to assign fuel types to 3 radial fuel zones in each axial segment if the input parameter IFUELC=1.
Block 61, GEOMIN			
224-247	RIZNC		Outer radius of central zone in each axial segment. The steady-state initialization routine makes this radius correspond to the input IZNC.
248-271	RIZNM		Outer radius of middle zone in each axial segment. The steady-state initialization routine makes this radius initialization routine makes this radius correspond to the input IZNM.
Block 65, FUELIN			
54	BURNFU		Axially averaged fuel pin burnup in atom percent.

10.4 Future Directions for Modeling Efforts

Multiple annular fuel zones of different compositions formed at the end of steady-state irradiation in U-Pu-Zr alloy fuel can not be supplied to the SAS4A/SASSYS-1 codes through the input data. The codes are capable of using the input zone formation data in the steady-state and transient temperature calculations. U-Pu-Zr fuel thermal properties can now be evaluated by the codes as a function of composition and temperature, using the method of regionwise interpolation of IFR Handbook data. Besides the U-Pu-Zr fuel specific heat, theoretical density and thermal conductivity, the codes can now evaluate the zonal solidus and liquidus temperatures and heat of fusion. The effect of radial migration of fuel constituents on in-pin radial power shape is yet to be modeled.

The purpose of the SSCOMP module is to provide SAS4A/SASSYS-1 codes with the modeling of all those phenomena that describe the transition of a U-Pu-Zr fuel pin from cold clean as-manufactured conditions to hot irradiated swollen initial conditions for the transition fuel mechanics module FPIN2 [10-6]. As described in Section 10.1, listed below are the important in-pin phenomena occurring during steady-state irradiation, that need to be modeled in order to characterize a U-Pu-Zr fuel pin at the beginning of a transient calculation:

1. Thermal expansion of fuel and cladding,
2. Fuel constituent radial migration,
3. Fission gas behavior, and porosity formation and distribution,
4. Irradiation-induced radial and axial swelling of fuel and cladding,
5. Bond sodium migration into fuel and pin plenum,

6. Cladding constituent migration into fuel.

The modeling of the above phenomena will provide all the needed initial conditions for the transient fuel mechanics module FPIN2. Future modeling efforts will be directed towards modeling the above phenomena. Detailed comments are made below about some of the above six phenomena.

10.4.1 Fuel constituent Radial Migration

As discuss in Section 10.2, the zone formation calculation method of the earlier SSCOMP model [10-4] is not incorporated into SAS4A/SASSYS-1 Version 3.0 codes due to its limitation. The study of fuel constituent radial migration leading to zone formation was not at that time developed to a state where an appropriate dynamic model could be developed. It is expected that a dynamic model for calculating fuel constituent radial migration, available in the literature or developed on the basis of available zonal composition and radial boundary data, will be incorporated into the codes. Such a dynamic model will be based on pin power history during steady-state irradiation. If neither a dynamic model nor a zonal composition and radial boundary database is available in the near future, the earlier SSCOMP model will be implemented in the codes.

Table 10.4.1: Internal Arrays Required for Implementing IFR Handbook U-Pu-Zr Fuel Properties Data and Multiple Radial Fuel Zones Option

Definition/Comments	Array Name
Arrays by Radial Node and Axial Segment in Block COMC	
1. Fuel mass	FUELMS(11,24)
2. Pu mass in fuel	FUPUMS(11,24)
3. Zr mass in fuel	FUZRMS(11,24)
4. Fe mass in fuel	FUFEMS(11,24)
5. Ni mass in fuel	FUNIMS(11,24)
6. Total porosity in fuel	PRSTY2(11,24)
7. Logged sodium mass in fuel	FUNAMS(11,24)
Arrays by Fuel Type in Block FPTVAR	
8. First solid-state transition temperature of fuel	TEMALF(8)
9. Last solid-state transition temperature of fuel	TEMGAM(8)
10. Pre-calculated coefficients in the quadratic equation for fuel thermal conductivity	ACPF(8,7)
11. Pre-calculated coefficient in the quadratic equation for fuel thermal conductivity	AKFU(8,5)
12. Pre-calculated parameters in fuel theoretical density equation	ADFU(8,2)

10.4.2 Fission Gas Behavior, and Porosity Formation and Distribution

The modeling of the following aspects of fission gas behavior is required for calculation the initial conditions of FPIN2 transient calculation: (a) fission gas generation, (b) fission gas retained with grains, (c) fission gas retained in grain boundary bubbles that are not linked, (d) fission gas contained in grain boundary bubbles that are interlinked, and (e) fission gas released to pin plenum. The need for modeling these three types of fission gas contained in fuel (gas within grains, in unliked grain boundary bubbles, and in interlined grain boundary bubbles) has also been expressed by the pre-failure in-pin fuel motion module PINACL development [10-7].

10.4.3 Irradiation-Induced Radial and Axial Swelling of Fuel and Cladding

Regarding irradiation-induced fuel swelling, the axial swelling will contribute to pre-transient fuel column length, a needed initial condition of the FPIN2 module. Radial swelling of fuel and cladding will determine fuel-cladding gap closure and bond sodium migration into pin plenum, a needed initial condition of the FPIN2 module. The modeling of fuel swelling phenomenon has two important aspects: swelling due to solid fission products, and fission gas induced swelling. Swelling of a cladding material is mainly determined by neutron fluence which is also a parameter in determining cladding rupture life used in the FPIN2 transient fuel mechanics calculation.

The SSCOMP module is being developed to model the above listed in-pin phenomena during steady-state irradiation of metallic fuel pins. After the fuel pin has been characterized at the end of steady-state irradiation, the calculation of the relevant ones of these phenomena during the transient will be performed by the DEFORM-5 model.

10.5 References

10-1. C. E. Till, “The IFR Status and Prospects,” *Trans. Am. Nucl. Soc.*, vol. 66, p. 351, November 1992.

10-2. J. H. Kittel, et al., “Plutonium and Plutonium Alloys as Nuclear Fuel Elements,” *Nucl. Eng. Design*, vol. 15, pp. 373-440, 1971.

10-3. W. F. Murphy, et al., “Post-irradiation Examination of U-Pu-Zr Fuel Elements Irradiated in EBR-II to 4.5 Atomic Percent Burnup,” ANL-7602, Argonne National Laboratory, Illinois, November 1969.

10-4. Kalimullah, K. J. Miles, Jr., A. M Tentner and D. P. Weber, Unpublished information, Argonne National Laboratory, 1985.

10-5. L. C. Walters, B. R. Seidel and J. H. Kittel, “Performance of Metallic Fuels and Blankets in Liquid-Metal Fast Breeder Reactors,” *Nucl. Technol.*, vol. 65, pp. 179-231, May 1984.

10-6. T. Sofu, personal communication, Argonne National Laboratory, January 1994.

10-7. A. M. Tentner, Unpublished information, Argonne National Laboratory, 1993.

10-8. “Handbook of Properties of Metallic Fuels,” compiled by IFR Property Evaluation Working Group, G. L. Hofman, Chairman, Argonne National Laboratory, Idaho Falls, Idaho, June 1988.

10-9. G. L. Hofman, et al., Unpublished information, Argonne National Laboratory, 1985.

10-10. R. Hultgren, et al., “Selected Values of the Thermodynamic Properties of the Elements,” *American Soc. For Metals*, Metals Park, Ohio, 1973.

10-11. J. M. Kramer, T. Hughes and E. E. Gruber, “Validation of Models for the Analysis of the Transient Behavior of Metallic Fast Reactor Fuel,” *Proc. of the 10:sup:`th` Intl. Conf. on Structural Mechanics in Reactor Technology*, Anaheim, CA, vol. C, pp. 65-76, August 14-18, 1989.

10-12. M. C. Billone, Unpublished information, Argonne National Laboratory, 1991.

10-13. L. Leibowitz and A. D. Pelton, personal communication, Argonne National Laboratory, September 1987.

10-14. "Reactor Handbook, Vol. 1, Materials," Second Edition, C. R. Tipton, Jr. Editor, Interscience Publishers, Inc., New York, 1960.

10-15. J. J. Sienicki, Unpublished information, Argonne National Laboratory, 1984.

10-16. Kalimullah and A. M. Tentner, Unpublished information, Argonne National Laboratory, 1988.

10-17. Kalimullah, Unpublished information, Argonne National Laboratory, 1993.

10-18. Y. S. Touloukian, et al., "Thermophysical Properties of Matter," Thermal Conductivity, Metallic Elements and Alloys, Vol. 1, IFI/Plenum, New York, 1970.

10-19. L. Leibowitz, Unpublished information, Argonne National Laboratory, 1984.

10-20. Argonne National Laboratory, Unpublished information, 1966.

10-21. M. C. Billone, Unpublished information, Argonne National Laboratory, 1991.

10-22. M. C. Billone, Unpublished information, Argonne National Laboratory, 1991.

10-23. "Plutonium Handbook, A Guide to the Technology," Vol. 1, O. J. Wick, Editor, Division of Technical Information, USAEC, Gordon and Breach, Science Publishers, London, 1967.

10-24. "Smithhells Metals Reference Book," 6th Edition, E. A. Brandes, Editor, Butterworths, Boston, 1983.

10-25. C. F. Beaton and G. F. Hewitt, "Physical Property Data for the Design Engineer," Hemisphere Publishing Corporation, Washington, 1989.

10-26. D. R. Olander, "Fundamental Aspects of Nuclear Reactor Fuel Elements," TID-26711-P1, pp. 124-127, Energy Research and Development Administration, NTIS, Springfield, Virginia, 1976.

10-27. R. A. di Novi, "Effect of Burnup, Swelling and Irradiation Temperatures on Thermal Diffusivity and Conductivity of Uranium-Fission Alloy," ANL-7889, Argonne National Laboratory, 1972.

10-28. W. N. Beck and R. J. Fousek, "Fission Gas Release and Thermal Conductivity Measurements on U-5 wt % Fs Irradiated in CP-5," *Trans. Am. Nucl. Soc.*, vol. 12, pp. 78-79, 1969.

10-29. P. R. Betten, "In-core Measurements of U-5 wt % Fissium Alloy Thermal Conductivity," *Trans. Am. Nucl. Soc.*, vol. 50, pp. 239-240, 1985.

10-30. R. A. Wigeland and T. Sofu, personal communication, Argonne National Laboratory, August 1993.

10-31. Kalimullah, M. C. Billone and J. M. Kramer, Unpublished information, Argonne National Laboratory, 1993.

10-32. L. L. Briggs, personal communication, Argonne National Laboratory, January 1993.

10.6 Nomenclature

Symbols	Definition	Units
A	Atom fraction, e.g., Eq. (10.3-7) or coefficients of the quadratic function of temperature for fuel thermal conductivity, e.g., Eq. (10.3-52) and Eq. (10.3-61).	
a	Atomic weight	
B	Coefficient of the quadratic function of temperature for fuel thermal conductivity, e.g., Eq. (10.7-12) to Eq. (10.7-14)	

continues on next page

Table 10.6.1 – continued from previous page

Symbols	Definition	Units
Bu	Fuel burnup	atom %
C	Plutonium alloying coefficient in thermal conductivity Eq. (10.3-102)	
C_{p}	Specific heat	J/kg-K
c	Coefficient of the quadratic term in the simplified enthalpy Eq. (10.3-26)	
D	Value of determinants in Cramer's rule for solving a system of linear algebraic equations	
f_{o}	Heavy-metal mass fissioned expressed as a percentage of U-Pu-Zr alloy post-irradiation mass	
H	Enthalpy	J/mol
h	Enthalpy	J/kg
K	Thermal Conductivity	W/m-K
K_{pt1}	Thermal conductivity of a pore tube containing a fission gas-filled pore	W/m-K
K_{pt2}	Thermal conductivity of a pore tube containing a logged sodium-filled pore	W/m-K
L	Length, e.g., Eq. (10.3-44) or Lorenz number in Q. Eq. (10.3-68)	m or W-Ω/K^2
M	Mass	kg
m_{a}	One atomic mass unit, $1.6592828 \times 10^{-27}$ kg	kg
N	Number of atoms, e.g., Eq. (10.3-34) or number of fissions, e.g., Eq. (10.3-35) or number of pairs of temperature and enthalpy values, e.g., Eq. (10.3-37)	
Na	Avogadro's number, 6.02472×10^{26} molecules/kg-mole	/kg-mole
P'	Porosity fraction relative to the 100%-dense volume	
P_{c1}	Areal fraction of a face of a unit cell that is occupied by a sodium-filled pore, e.g., Eq. (10.3-72)	
P_{c2}	Areal fraction of a face of a unit cell that is occupied by a sodium-filled pore, e.g., Eq. (10.3-72)	
P_{g}	Gas-filled porosity fraction relative to the swollen fuel volume	
$P_{\mathrm{l,1}}$	Fraction of the side (in heat flow direction) of a unit cell that is occupied by a gas-filled pore	
$P_{\mathrm{l,2}}$	Fraction of the side (in heat flow direction) of a unit cell that is occupied by a sodium-filled pore	
P_{Na}	Sodium-filled porosity fraction relative to the swollen fuel volume	
T	Temperature	K
V	Volume	m^3
v	Characteristic volume of an atom	m^3
W	Weight fraction	
X_{i}	Weight fractions of the 3 database alloys to be mixed to produce the desired fuel composition	
x	Zr weight fraction in the U-Zr binary database alloy used in enthalpy region 1 or thermal conductivity region 3	
Y_{i}	Atom fractions of the 3 database alloys to be mixed to produce the desired fuel composition	
α	Coefficient of linear thermal expansion	per K
ΔL	Linear thermal expansion from 293K	m
ρ	Density, e.g., Eq. (10.3-32) or electrical resistivity in Eq. (10.3-68)	kg/m^3 or Ω-m
σ	Standard deviation	
Subscripts:		
a	Desired alloy or desired database alloy	

continues on next page

Table 10.6.1 – continued from previous page

Symbols	Definition	Units
c	Centigrade temperature scale	
f	Fission, e.g., Eq. (10.3-38) or solid fission products, e.g., Eq. (10.3-40) or desired fuel, e.g., Eq. (10.3-61)	
g	Fission gas	
h	Heavy metals	
i	Index for fuel constituents (I - U, Pu, Zr) or index for the 3 database alloys of a region of interpolation	
l	Liquidus or molten liquid	
Na	Sodium	
o	Fabricated condition, e.g., Eq. (10.3-33) or 100%-dense unirradiated condition, e.g., Eq. (10.3-72) or a specified temperature, e.g., Eq. (10.3-44)	
p	Plutonium	
s	Solidus	
sfp	Solid fission products	
th	Theoretical	
u	Uranium	
uz	U-Zr binary alloy	
u50	U-50 atom % Zr binary alloy	
z	Zirconium	
α	Lowest-temperature solid-state phase (α-phase)	
γ	Highest-temperature solid-state phase (γ-phase)	

10.7 Appendices

10.7.1 APPENDIX 10.1: Equivalency of Interpolation in Terms of Weight Fractions and Atom Fractions

As described in Section 10.3.2.1, the enthalpy of a desired U-Pu-Zr fuel composition is evaluated by mixing 3 basic alloys with enthalpy data known and reported in the Metallic Fuels Handbook [10-8]. The proportions or weight fractions of the 3 database alloys in mixing are so calculated that the resulting mixture has the composition of the desired fuel. The purpose of this appendix is to show that it makes no difference whether this mixing or linear interpolation is carried out by expressing composition in terms of weight fractions or atom fractions, and correspondingly enthalpy in terms of J/unit mass or J/mol.

Let

a_1, a_2, a_3 = average atomic weights of the 3 database alloys,

h_1, h_2, h_3 = enthalpy (J/gm) of the 3 database alloys at a desired temperature.

H_1, H_2, H_3 = enthalpy (J/mol) of the 3 database alloys at the desired temperature,

X_1, X_2, X_3 = weight fractions of the 3 database alloys to be mixed to produce the desired fuel composition.

Y_1, Y_2, Y_3 = atom fractions of the database alloys to be mixed to produce the desired fuel composition.

The desired fuel enthalpy interpolated in terms of weight fractions is given by

$$h = X_1 h_1 + X_2 h_2 + X_3 h_3 \tag{10.7-1}$$

where h is in units of J/gm. The desired fuel enthalpy interpolated in terms of atom fractions is given by

$$H = Y_1 H_1 + Y_2 H_2 + Y_3 H_3 \tag{10.7-2}$$

where H is in units of J/mol. The enthalpies given by Eq. (10.7-1) and Eq. (10.7-2) will be the same if it is shown that

$$h = \frac{H}{a} \tag{10.7-3}$$

where

a = average atomic weight of the desired fuel composition.

The average atomic weight of the desired fuel composition is related to the atomic weight of the 3 database alloys and their weight fractions in the desired fuel as follows:

$$a = \frac{1}{\left(\frac{X_1}{a_1} + \frac{X_2}{a_2} + \frac{X_3}{a_3}\right)} \tag{10.7-4}$$

The atom fraction Y_1 of database alloy 1 in the desired fuel is related to the weight fractions X_1, X_2, X_3 of the 3 database alloys in the desired fuel as follows:

$$Y_1 = \frac{\left(\frac{X_1}{a_1}\right)}{\left(\frac{X_1}{a_1} + \frac{X_2}{a_2} + \frac{X_3}{a_3}\right)} \tag{10.7-5}$$

With the help of Eq. (10.7-4), the atom fraction Y_1 given by Eq. (10.7-5) simplifies to

$$Y_1 = \frac{aX_1}{a_1} \tag{10.7-6}$$

Similarly, the atom fractions Y_2 and Y_3 can be written as

$$Y_2 = \frac{aX_2}{a_2} \tag{10.7-7}$$

$$Y_3 = \frac{aX_3}{a_3} \tag{10.7-8}$$

The enthalpies of the 3 database alloys in units of J/mol are related to their enthalpies in units of J/gm and their atomic weights as follows:

$$H_1 = a_1 h_1, \quad H_2 = a_2 h_2, \quad H_3 = a_3 h_3 \tag{10.7-9}$$

Substituting the values of Y_1, Y_2, Y_3 and H_1, H_2, H_3 from Eq. (10.7-6) to Eq. (10.7-9) into Eq. (10.7-2), one obtains

$$H = aX_1 h_1 + aX_2 h_2 + aX_3 h_3 \tag{10.7-10}$$

With the help of Eq. (10.7-1), the desired fuel enthalpy interpolated in terms of atom fractions given by Eq. (10.7-10) becomes

$$H = ah \tag{10.7-11}$$

Eq. (10.7-11) proves what Eq. (10.7-3) required.

10.7.2 APPENDIX 10.2: Correlations in the IFR Handbook and Some Other Sources

The following 4 correlations are available to estimate 100% dense solid U-Pu-Zr fuel thermal conductivity, K_o (W/m-K):

1. The correlation in the IFR Metallic Fuels Handbook [10-8, 10-9] is written as

$$K_o = B_1 + B_2 T + B_3 T^2 \tag{10.7-12}$$

where

$B_1 = 17.5 \left[\frac{(1-2.23W_z)}{(1+1.61W_z)} - 2.62W_p \right]$

$B_2 = 0.0154 \left[\frac{(1+0.061W_z)}{(1+1.61W_z)} + 0.90W_p \right]$

$B_3 = 9.38 \times 10^{-6} (1 - 2.70W_p)$

W_p, W_z = weight fractions of Pu and Zr

T = fuel temperature, K.

2. The correlation in Billone's memorandum [10-21] of March 8, 1991 is written as

$$K_o = B_4 + B_5 T + B_6 T^2 \tag{10.7-13}$$

where

$B_4 = 17.5 \left[1 - 1.471A_z + 44.083A_p \left(1 - 16.198A_p + 55.290A_p^2 \right) \right]$

$B_5 = 0.0154 \left[1 - 0.5935A_z - 158.88A_p \left(1 - 14.865A_p + 49.478A_p^2 \right) \right]$

$B_6 = 9.38 \times 10^{-6} \left[1 + 215.78A_p \left(1 - 14.421A_p + 47.158A_p^2 \right) \right]$

A_p, A_z = atom fractions of Pu and Zr.

3. The correlation in Billone's memorandum [10-22] of October 21, 1991 is written as

$$K_o = B_7 + B_8 T + B_9 T^2 \tag{10.7-14}$$

where

$B_7 = 17.5 (1 - 1.417A_z - 6.8174A_p)$

$B_8 = 0.0154 (1 - 0.5935A_z + 15.108A_p)$

$B_9 = 9.38 \times 10^{-6} (1 - 18.778A_p)$

4. The correlation reported in ANL/RAS 85-19 can be written as

$$K_o = K_u + 2A_z \frac{(K_{u50} - K_u)}{(A_u + A_z)} - CA_p \tag{10.7-15}$$

where

$K_u = 22.173 + 0.018562T_c + 1.3278 \times 10^{-5} T_c^2$

$K_{u50} = 7.4766 + 0.016738T_c + 1.1599 \times 10^{-5} T_c^2$

$C = (89.079 - 167.57A_z) - T_c (0.17437 - 0.4008A_z) + T_c^2 \left(1.9608 \times 10^{-4} - 3.1043 \times 10^{-4} A_z \right)$, $100 \leq T_c \leq 1500$,

$$0.0 \leq A_p \leq 0.2, \quad 0.0 \leq A_z \leq 0.5 \tag{10.7-16}$$

T_c = fuel temperature, °C

Correlations 1, 2 and 3 have been developed based on the thermal conductivity data presented in the Metallic Fuels Handbook. Detailed plots of these correlations are available in Ref. [10-17]. Correlation 2 should not be used because it is ill-behaved, and so recommended by Billone [10-22]. A drawback of correlations 1 and 3 is that the thermal conductivities become negative (unphysical) at room temperature for Pu and Zr contents greater than 20 wt %. In addition, the thermal conductivity obtained from correlation 3 shows a maximum when plotted as a function of temperature (an unexpected behavior [10-18] for temperatures above 300 K).

The comparison of these 3 correlations with the Handbook data for 3 ternary alloys (U-16.2Pu-6.2Zr, U-14.7Pu9.7Zr and U-18.4Pu-11.5Zr) shows that there are significantly large differences among the 3 correlations at temperatures above 1200 K where there is no thermal conductivity data. Below 1200 K, the difference among the data and the 3 correlations are smaller.

The comparison of these 3 correlations with the Handbook data for 3 U-Zr binary alloys (U-1.5Zr, U-5Zr and U-20Zr) shows that the 3 correlations are almost identical at all temperatures, and that they agree with the data closely. The agreement is not so good with the data for U-40Zr, as shown by a plot given in the Handbook [10-8, 10-9]. The Handbook correlation for U-Zr binary alloys, i.e., Eq. (10.7-12) with W_p set to zero, is found to be reliable for Zr content in the 1.5 to 20 wt % range.

The thermal conductivity obtained from correlation 4 for the some alloy compositions becomes negative (unphysical) at room temperature, and shows an unexpected maximum when plotted as a function of temperature. Although the alloy compositions showing this unphysical/unexpected behavior are out of the noted region of validity, the correlation has another drawback that it was derived using data from sources [10-14, 10-19, 10-20] other than the IFR Metallic Fuels Handbook [10-8, 10-9].

CHAPTER

ELEVEN

FPIN2: PRE-FAILURE METAL FUEL PIN BEHAVIOR MODEL

11.1 Introduction

Since fuel element behavior during fast reactor transients can significantly affect accident energetics, an accurate and detailed model for the mechanical response of fuel elements is needed in safety assessments. During transients that lead to overheating of fuel and cladding, plastic straining of the cladding due to internal fission gas pressure and differential expansion between the fuel and cladding may result in cladding rupture and release of fission products to the primary coolant. In addition to the fundamental concern for cladding failure, fuel axial expansion can also have an important role due to the associated negative reactivity potential.

FPIN2 is a validated computer code that provides mathematical models that simulate fuel and cladding mechanical response and predict fuel element performance over a wide range of transients. It performs an analysis of metal fuel and cladding deformation, including the impact of fuel-cladding interactions, and estimates cladding failure locations and times.

FPIN2 has been incorporated into the SAS4A/SASSYS-1 code system for mechanical analysis of individual fuel elements. In this implementation, SAS4A/SASSYS-1 provides fuel and cladding temperatures, and FPIN2 performs the analysis of fuel element deformation and predicts the time and location of cladding failure. FPIN2 results are also used for the estimates of axial fuel expansion and the associated reactivity effects. In this chapter, FPIN2's mechanical model, the SAS-FPIN2 coupling methodology, and the integrated SAS-FPIN2 model usage are presented.

11.1.1 Brief Description of FPIN2

The FPIN2 code has been developed to analyze the thermal-mechanical phenomena that control fuel element behavior during fast reactor transients. The early versions of the code were based on the characteristics of oxide and carbide ceramic fuels [11-1,11-2]. More recently, FPIN2 has been adapted for the analysis of metallic fuels [11-3,11-4]. The overall modeling for this metallic fuel version of the code was validated through comparison of FPIN2 calculations with the data from TREAT tests on EBR-II irradiated fuel, prototypic of the IFR concept [11-5,11-6]. The most recent version of the code integrated with SAS4A/SASSYS-1 includes numerous model improvements that reflect the experience gained during these validation efforts.

A wide range of material behavior is modeled in the code that describes elastic, plastic, thermal, and swelling performance of fuel elements. Since the primary emphasis in FPIN2 is on the mechanical analysis of the fuel and cladding, in the standalone version of the code temperatures are calculated using a simple model based on pin-in-a-pipe geometry and single-phase flow. The mechanical model, on the other hand, is based on a rigorous force-displacement formulation and uses an implicit finite element method with linear shape functions. The finite element scheme used allows convenient modular coding in which different models for material behavior and improvements in specific algorithms are easily implemented. The equilibrium equations are derived form equations of virtual work. The elements are defined in an (r,z) mesh; however, axial symmetry and generalized

plane strain are assumed so that the analysis is essentially one-dimensional. The elements are allowed to interact only at the radial boundaries (nodes), and the displacements within the elements are approximated by linear functions of the nodal displacements.

Additional models for metallic fuels such as models for fission gas generation and release, molten cavity formation, the large gas plenum, and fuel-cladding eutectic alloy formation are also provided in FPIN2 to complement the fuel element mechanics calculations. Internal pin pressure is determined from direct mass and volume balances in the central cavity and gas plenum. The molten fuel cavity in FPIN2 is located by the axial and radial extent to which the fuel has reached its solidus temperature and the elements inside the cavity boundary are dropped out of the stress-chain calculation. For cases where initial fuel melting occurs below the top axial segment, the plenum pressure calculation is decoupled from the molten cavity pressure calculation. Once melting reaches cavity-gas plenum interface, the two pressures are assumed to equilibrate and the plenum/cavity pressure-volume equations are solved together to give a common pin pressure and amount of molten fuel extruded into the plenum.

Checking rupture is predicted in the code by using the life fraction criterion. The life fraction change over a time step is determined from the change in rupture time for the instantaneous average cladding temperature and the hoop stress obtained form the thin-shell equations used in developing the life fraction correlations. The effect of low melting point eutectic formation between the fuel and the cladding is included in the calculations by considering only the thickness of unaffected cladding that is available to carry the load.

11.1.2 Overview

The finite element formulation that constitutes the basic structure of FPIN2's mechanics calculation is described in Section 11.2. Although the models presented in Section 11.2 are primarily developed for oxide fuel, they provide a substantially robust structure that allows modifications to handle metallic fuels. The modifications and additions to the code for metal fuels are discussed separately in Section 11.3. Most of these models for metallic fuels reflect the experience gained during FPIN2 validation efforts through comparison with the data from TREAT tests.

Two modes of SAS-FPIN2 coupled operations are provided. In the stand-alone mode, independent FPIN2 input is used and FPIN2 is executed without interfacing to SAS4A/SASSYS-1. In the interfaced mode, SAS4A/SASSYS-1 calculated fuel and cladding temperatures are transferred to FPIN2, and FPIN2 results are used in the analysis of fuel behavior. The SAS-FPIN2 coupling methodology and the integrated SAS-FPIN2 model usage along with input and output descriptions are presented in Section 11.4.

11.2 Fuel Element Mechanics

The mechanical model of FPIN2 uses an implicit finite element method based on a force-displacement formulation. The elements are defined by dividing the fuel and cladding into a number of axial segments and radial rings. Axial symmetry of the loads and generalized plane strain are assumed; therefore, the elements are allowed to interact only at the radial boundaries (nodes). The displacements within the elements are approximated by linear functions.

The mechanical analysis of FPIN2 basically provides the fuel and cladding stresses, σ, and nodal displacements, δ, for each element during overpower and undercooling events. The fuel and cladding behavior model includes thermal expansions, elastic/plastic/creep deformations, cracking, and melting. A continuous cracking model is used in which an element is assumed to crack in the radial direction when the tensile stress exceeds the fracture stress. Pre-existing cracks are also handled by specifying initial crack strains.

The finite element equilibrium equations are derived from equations of virtual work that leads to a force balance

between the external and internal forces

$$\underline{f}^{\text{int}}(\sigma) = \underline{f}^{\text{ext}} \tag{11.2-1}$$

The external source vector in this equation, $\underline{f}^{\text{ext}}$, describes the external loads at the cavity boundary and the cladding outer surfaces, and the internal force vector, $\underline{f}^{\text{int}}(\sigma)$, depends on the unknown stress field that must be found to satisfy Eq. (11.2-1) and the constitutive laws for the materials. When this generally non-linear set is linearized and the equations for both the fuel and cladding elements are assembled together for the unknown nodal displacements at each axial segment, the resulting linear equation set can be written symbolically as

$$[K]\,\underline{\delta} = \underline{f} \tag{11.2-2}$$

where $[K]$ is the stiffness matrix. An elastic-plastic approximation is used in FPIN2 to derive the elements of the stiffness matrix involving the total derivatives of the constitutive equations. The force vector in Eq. (11.2-2) includes the external loads, thermal and other initial strains, and pseudo-forces involving approximations to the plastic and/or creep strains.

The equilibrium equations, Eq. (11.2-1), and the linear equations, Eq. (11.2-2), are solved iteratively for each axial segment by making use of residuals. In this scheme, first the stiffness matrix, $[K]$, and an initial guess for the force vector, $\underline{f}_0$, are assembled using available information from the previous time step. Then, the linear equations Eq. (11.2-2) are solved for the new nodal displacements, and from them the increments of total strain are obtained. Assuming that the total material strain is a superposition of elastic, plastic, thermal, and swelling strains, the elastic component of the increments of total strain is separated (see Section 11.2.2) and used to calculate the stress field throughout the fuel element that satisfies the material constitutive equations. These stresses, then, are used to obtain the residuals form the equilibrium equations, Eq. (11.2-1), as follows

$$\underline{r}_{\text{n}} = \underline{f}^{\text{ext}} - \underline{f}^{\text{int}}(\sigma_{\text{n}}) \tag{11.2-3}$$

where n is the iteration index. Using these residuals, the initial estimate for the force vector in linear equations Eq. (11.2-2) is modified according to

$$\underline{f}_{\text{n}+1} = \underline{f}_{\text{n}} + \underline{r}_{\text{n}} \tag{11.2-4}$$

The procedure is repeated until convergence is established. This basic scheme is generalized in FPIN2 to include analysis of additional material performance in fuel and cladding such as cracking and melting.

The details of this procedure described above are given in the following sections. The basic stress, strain, and nodal displacement relationships are discussed in Section 11.2.1, and the algorithm for separating elastic strain from the total strain is presented in Section 11.2.2. FPIN2 uses a continuous cracking model that permits cracks in the radial plane. The principles of this model are described in Section 11.2.3. The finite element formulation and the details of the implicit solution scheme outlined above are presented in Section 11.2.4. The formulations given in the aforementioned sections are based on updated geometry and can handle large strains. This is accomplished by performing the mechanics calculation on an incremental basis and updating the fuel element geometry at the end of each time step as described in Section 11.2.5.

11.2.1 Basic Stress-Strain-Displacement Equations

In this section, a number of definitions and basic equations relating stresses, strains, and modal displacements are collected and presented in a uniform notation. These equations constitute the foundation of FPIN2's mechanics model and serve as the basis for derivations in later sections. Formulas are derived for large-strain finite element analysis of deformation in a cylindrical fuel element under accident conditions. At the beginning of the analysis, the fuel element is assumed to be stress-free with steady-state thermal conditions. Thermoeleastic-plastic behavior is assumed for both the fuel and cladding. Swelling (including hot-pressing) and cracking are also allowed in both materials.

Fuel elements are assumed to consist of a number of axial segments. The finite elements are defined in each axial segment separately and allowed to interact only at the radial boundaries (nodes). The displacements within the elements are approximated by linear functions of the nodal displacements. Axial symmetry of temperatures and external loads is assumed; therefore, the finite elements can be viewed as a series of concentric annular rings. The generalized plane strain assumption is used describe the interaction between the axial segments.

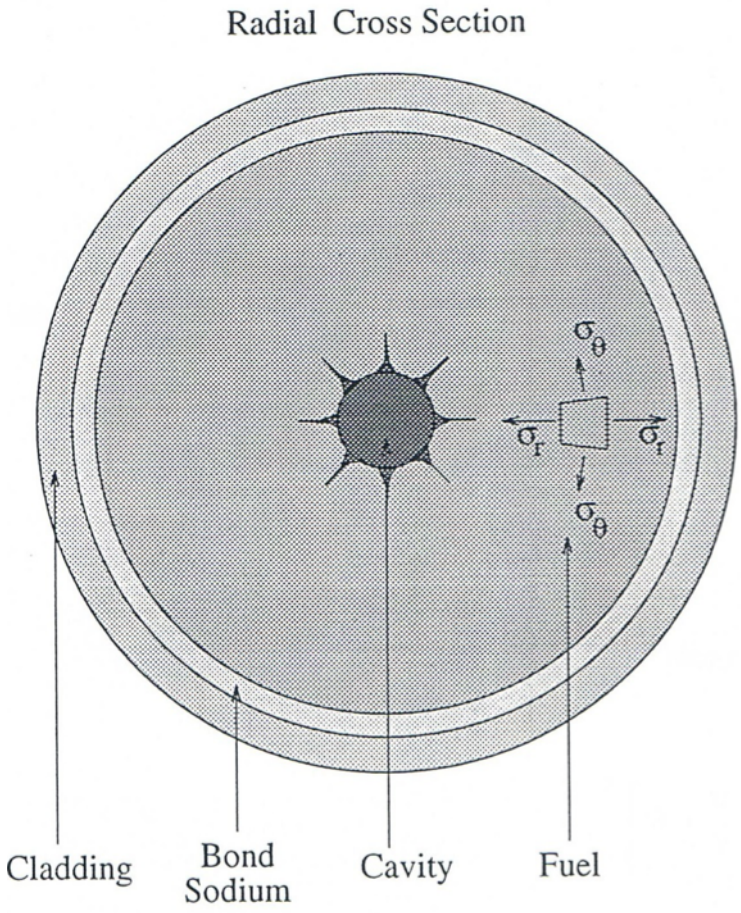


Figure 11.2.1: FPIN2 Geometry

Each axial segment is allowed to have regions containing central cavity gas and/or molten fuel, regions of solid material that are uncracked, and regions that contain radial cracks (Figure 11.2.1). If an element does not contain any cracks, the axisymmetry assumption dictates that the deformation depends only on r. Although the existence of radial cracks, in general, destroys this symmetry, it is assumed that the number of cracks in an element is large enough to make the deformation essentially axisymmetric.

11.2.1.1 Strain-Displacement Relationships

In cylindrical coordinates, the strain-displacement relations are given by

$$\underline{\varepsilon}^t = \left\{\varepsilon_{\mathrm{r}}^t, \varepsilon_\theta^t, \varepsilon_{\mathrm{z}}^t\right\}^T = \left\{\frac{\partial \mathrm{u}}{\partial \mathrm{r}}, \frac{u}{r} + \frac{1}{r}\frac{\partial \mathrm{v}}{\partial \theta}, \frac{\partial \mathrm{w}}{\partial \mathrm{z}}\right\}^T \tag{11.2-5}$$

where u, v, and w are displacements in r, θ and z coordinate directions, respectively. If the material is continuous, the deformation depends only on r due to axisymmetry of temperatures and loads. If the material contains radial cracks, the tangential displacement, v, can be expressed as a linear function of θ in a first order approximation assuming that sufficiently numerous radial cracks exist in the element. In the axial direction, the generalized plane strain condition requires that

$$w = \psi z \tag{11.2-6}$$

where ψ is a constant.

For convenience, we may rewrite the strain-displacement relation as follows to differentiate the crack strain from continuous material strain

$$\underline{\varepsilon}^t = \underline{\varepsilon}^a + \underline{\varepsilon}^c \tag{11.2-7}$$

where "apparent total strain" and "crack strain" vectors are given respectively by

$$\underline{\varepsilon}^a = \left\{\frac{\mathrm{du}}{\mathrm{dr}}, \frac{u}{r}, \psi\right\}^T \tag{11.2-8}$$

$$\underline{\varepsilon}^c = \left\{0, \varepsilon_\theta^c, 0\right\}^T \tag{11.2-9}$$

Here, $\underline{\varepsilon}_\theta^c$ is the unknown crack strain component related to radial creaks in the material.

11.2.1.2 Components of Total Strain

The total material strain, $\underline{\varepsilon}^t$, can be written as the sum of the elastic strain, $\underline{\varepsilon}^e$, the plastic strain, $\underline{\varepsilon}^p$, the thermal strain, $\underline{\varepsilon}^\theta$, and the swelling strain, $\underline{\varepsilon}^s$:

$$\underline{\varepsilon}^t = \underline{\varepsilon}^e + \underline{\varepsilon}^p + \underline{\varepsilon}^\theta + \underline{\varepsilon}^s \tag{11.2-10}$$

11.2.1.2.1 Elastic Strain

In linear elasticity, the components of elastic strain are related to normal stresses through the generalized Hook's law

$$\underline{\sigma} = [C]\,\underline{\varepsilon}^e \tag{11.2-11}$$

where the matrix $[C]$ is given in terms of Young's modulus, E, and Poisson's ratio, ν, as follows

$$[C] = \frac{E\,(1-\nu)}{(1+\nu)\,(1-2\nu)} \begin{bmatrix} 1 & \frac{\nu}{(1-\nu)} & \frac{\nu}{(1-\nu)} \\ \frac{\nu}{(1-\nu)} & 1 & \frac{\nu}{(1-\nu)} \\ \frac{\nu}{(1-\nu)} & \frac{\nu}{(1-\nu)} & 1 \end{bmatrix} \tag{11.2-12}$$

In general, the elastic parameters E and ν are functions of temperature, T.

11.2.1.2.2 Thermal Strain

In the original oxide fuel version of FPIN2, the components of isotropic thermal strain, $\underline{\varepsilon}^{\theta}$, are expressed in terms of mean thermal expansion coefficient, α_{m}, as follows

$$\varepsilon_{\mathrm{r}}^{\Theta} = \varepsilon_{\theta}^{\Theta} = \varepsilon_{\mathrm{z}}^{\Theta} = \alpha_{\mathrm{m}} \left(T - T_0\right) \tag{11.2-13}$$

where T_0 is a reference temperature. In more recent applications for metallic fuels, the thermal expansion is expressed in terms of handbook linear thermal expansion data [11-7], $\Delta L/L_0$, as discussed in Section 11.3.3. Using the thermal data provides more detailed information since it automatically includes the total fuel expansion at solid-to-solid phase transitions. Any other stress-independent isotropic swelling strains (such as selling due to solid fission products or small surface tension dominated gas bubbles) are treated in the same manner as thermal strains in the model.

11.2.1.2.3 Swelling strain

Stress dependent swelling strains, $\underline{\varepsilon}^{s}$, due to fission gas in the fuel are also assumed to be isotropic and required to be expressed by an implicit function of the form

$$g\left(\sigma_{\mathrm{m}}, \varepsilon^{s}, \frac{\mathrm{d}\varepsilon^{s}}{\mathrm{dt}}, T, \nabla T, \rho\right) = 0 \tag{11.2-14}$$

Here, the mean hydrostatic stress σ_{m} is calculated from

$$\sigma_{\mathrm{m}} = \frac{\sigma_{\mathrm{r}} + \sigma_{\theta} + \sigma_{\mathrm{z}}}{3} \tag{11.2-15}$$

Hot-pressing is also considered in the model as a negative contribution to the total swelling.

11.2.1.2.4 Plastic strain

The plastic deformation of the material is postulated to occur when a generalized *von Mises* flow condition is satisfied:

$$f\left(\sigma_{\mathrm{e}}, \overline{\varepsilon}^{p}, \frac{d\overline{\varepsilon}^{p}}{\mathrm{dt}}, T\right) = 0 \tag{11.2-16}$$

where the equivalent stress, σ_{e}, and equivalent plastic strain, $\overline{\varepsilon}^{p}$, are given respectively by the following equations

$$\sigma_{\mathrm{e}} = \sqrt{\frac{3}{2}\left(S_{\mathrm{r}}^2 + S_{\theta}^2 + S_{\mathrm{z}}^2\right)} \tag{11.2-17}$$

$$\overline{\varepsilon}^{p} = \int d\overline{\varepsilon}^{p} = \int \sqrt{\frac{2}{3}\left[\left(d\varepsilon_r^p\right)^2 + \left(d\varepsilon_\theta^p\right)^2 + \left(d\varepsilon_z^p\right)^2\right]} \tag{11.2-18}$$

with the deviatoric stresses defined as

$$S_{\mathrm{i}} = \sigma_{\mathrm{i}} - \sigma_{\mathrm{m}} \tag{11.2-19}$$

$(i = r, \theta, z)$. The normality condition of plasticity states that the increment of the plastic strain is proportional to the gradient in stress space of the flow condition:

$$\underline{\mathrm{d}\varepsilon}^{p} = \mathrm{d}\lambda \underline{W} \tag{11.2-20}$$

where $\mathrm{d}\lambda$ is the proportionality factor. The gradient vector is given by

$$\underline{W} = \left\{ \frac{\partial \sigma_{\mathrm{e}}}{\partial \sigma_{\mathrm{r}}}, \frac{\partial \sigma_{\mathrm{e}}}{\partial \sigma_{\theta}}, \frac{\partial \sigma_{\mathrm{e}}}{\partial \sigma_{\mathrm{z}}} \right\}^{T} = \frac{3}{2\sigma_{\mathrm{e}}} \left\{ S_{\mathrm{r}}, S_{\theta}, S_{\mathrm{z}} \right\}^{T} \tag{11.2-21}$$

For the particular form of flow conditions chosen in this model, the factor of proportionality, $\mathrm{d}\lambda$, is identical to the increment in equivalent plastic strain, $\mathrm{d}\bar{\varepsilon}^{p}$. The plastic strains are assumed to occur with zero volume change:

$$\varepsilon_{\mathrm{r}}^{p} + \varepsilon_{\theta}^{p} + \varepsilon_{\mathrm{z}}^{p} = 0 \tag{11.2-22}$$

The constitutive equations for plastic deformation as well as swelling (Eq. (11.2-16) and Eq. (11.2-14)) indicate that, in general, the deformation depends on the changes in stress and temperature. Also, a major part of the deformation in a fuel element at high temperatures is rate dependent. The discussion of the material properties for the swelling and plastic behavior of the metallic fuel pins is given in Section 11.3.

11.2.1.3 Incremental Stress-Strain Relationships

Since the elastic constants of the matrix $[C]$ in Eq. (11.2-11) are functions of temperature, T, the incremental elastic stress components are given by

$$\underline{\mathrm{d}\sigma} = [C]\,\underline{\mathrm{d}\varepsilon}^{e} + \frac{d\,[C]}{\mathrm{dT}}\underline{\varepsilon}^{e}\mathrm{dT} = [C]\left(\underline{\mathrm{d}\varepsilon^{e} - \frac{\mathrm{d}\,[C]^{-1}}{\mathrm{dT}}\sigma\mathrm{dT}}\right) \tag{11.2-23}$$

Using the equation representing the superposition of strains:

$$\underline{\mathrm{d}\varepsilon}^{e} = \underline{\mathrm{d}\varepsilon}^{t} - \underline{\mathrm{d}\varepsilon}^{p} - \underline{\mathrm{d}\varepsilon}^{\theta} - \underline{\mathrm{d}\varepsilon}^{s} \tag{11.2-24}$$

and replacing $\underline{\mathrm{d}\varepsilon}^{p}$ with equivalent plastic strain increments, the change in stress can be expressed in terms of increment of total strain as follows

$$\underline{\mathrm{d}\varepsilon} = [C]\left(\underline{\mathrm{d}\varepsilon}^{t} - \underline{W}d\bar{\varepsilon}^{p} - \underline{Q}\mathrm{d}T - \underline{\mathrm{d}\varepsilon}^{s}\right) \tag{11.2-25}$$

where, for convenience, the thermal modulus $\underline{Q}$ has been introduced:

$$\underline{Q} = \underline{\alpha} + \frac{\mathrm{d}\,[C]^{-1}}{\mathrm{dT}}\underline{\sigma} \tag{11.2-26}$$

The mechanical model of FPIN2 makes use of Eq. (11.2-25) to calculate approximate changes in stress given the estimates of plastic and swelling strain increments.

11.2.1.4 Element Types

The actual form of Eq. (11.2-25) depends on the type of fuel element considered. Originally, four different types of elements are considered in the finite element formulation of FPIN2 that allows analysis of cracks in both radial and transverse planes. However, only two types of elements are available in the metal fuel version because cracks in metallic fuels, if they occur, are predominantly radial. The definitions of the two types are as follows:

Type I: The element lies in continuous fuel or cladding. Both the axial and tangential crack strains are zero; therefore, the total stain in the element is equal to the apparent total strain. The *resultant* of the axial stress is also assumed to be known.[4]

[4] If the fuel-cladding gap is closed, this resultant axial force may include both fuel and cladding.

Type II: The element lies in a portion of the fuel where only radial cracks are present and the tangential crack strain is non-zero. The circumferential stress is constant everywhere in the element and equal to the boundary value $\sigma_\theta = -p_g$ (p_g is the pressure of the fission gas present in the crack opening.) As in Type I, the resultant axial force is assumed to be known.

Table 11.2.1: Source of Calculated Stress and Strain Results for the Element Types

Element Type	Equilibrium and Strain-Displacement Equations	Crack Strain Definition	Stress Boundary Conditions	Algorithm Described in Section 11.2.2	
I	Continuous	$\varepsilon_r^a, \varepsilon_\theta^a, \varepsilon_z^a$	$\varepsilon_\theta^c = 0$	—	$\sigma_r, \sigma_\theta, \sigma_z$
II	Radial cracks	$\varepsilon_r^a, \varepsilon_\theta^a, \varepsilon_z^a$	$\varepsilon_\theta^c \neq 0$	$\sigma_\theta = -p_g$	$\sigma_r, \sigma_z, \varepsilon_\theta^c$

In each case, solving the finite element equilibrium equations gives the radial displacement, $u\,(r)$, from which apparent strain, $\underline{\varepsilon}^a$, can be calculated. The stress and strain components not obtained from the equilibrium equations, strain-displacement relations, or stress boundary conditions are calculated by an algorithm to separate elastic strain from the total strain as described in Section 11.2.2 (see Table 11.2.1).

11.2.1.5 Relationships for Stress Increments: Pseudo-Stress

Since the actual form of Eq. (11.2-25) depends on the type of fuel element considered, it can be expressed in following generalized form

$$\underline{\mathrm{d}\sigma} = [C]\,\underline{\mathrm{d}\varepsilon}^a - \underline{\mathrm{d}\sigma}^{\mathrm{ps}} \tag{11.2-27}$$

where $\underline{\mathrm{d}\sigma}^{\mathrm{ps}}$ is the incremental pseudo-stress. Ad described in Section 11.2.4.2, given the estimates of plastic and swelling strain increments, this pseudo-stress term is treated as a constant within a time step. In other words, Eq. (11.2-27) expresses the changes in stress in terms of a known term $\underline{\mathrm{d}\sigma}^{\mathrm{ps}}$ and an unknown increment $\underline{\mathrm{d}\varepsilon}^a$. The calculated stress increments are then corrected to satisfy the equilibrium and constitutive equations to a desired accuracy as described in Section 11.2.4.2.

Existence of radial cracks in the element alters the form of the elastic modulus, $[C]$, and the pseudo-stress vector, $\underline{\mathrm{d}\sigma}^{\mathrm{ps}}$. As shown in Table 11.2.1, for an uncracked element all three components of the total apparent strain are to be found from the equilibrium equations. Therefore, the incremental stress can be written in the form of Eq. (11.2-27) as

$$\underline{\mathrm{d}\sigma} = [C]\,\underline{\varepsilon}^a - \underline{\mathrm{d}\sigma}_{\mathrm{I}}^{\mathrm{ps}} \tag{11.2-28}$$

where $[C_{\mathrm{I}}]$ and the incremental pseudo-stress are simply defined by

$$[C_{\mathrm{I}}] = [C] \tag{11.2-29}$$

and

$$\underline{\mathrm{d}\sigma}_{\mathrm{I}}^{\mathrm{ps}} = [C]\left(\underline{Q}\mathrm{dT} + \underline{W}\mathrm{d}\bar{\varepsilon}^p + \underline{\mathrm{d}\varepsilon}^s\right) \tag{11.2-30}$$

For an element with radial cracks (Type II), Table 11.2.1 indicates that

$$\underline{\mathrm{d}\varepsilon}^c = \{0, d\varepsilon_\theta^c, 0\}^T \tag{11.2-31}$$

$$\underline{\mathrm{d}\sigma} = \left\{\mathrm{d}\sigma_r, -\mathrm{dp}_g, \mathrm{d}\sigma_z\right\}^T \tag{11.2-32}$$

By substituting these equations into Eq. (11.2-25) and premultiplying by $[C]^{-1}$, $\mathrm{d}\varepsilon_\theta^c$ can be eliminated and the equations can be solved for $\mathrm{d}\sigma_\mathrm{r}$ and $\mathrm{d}\sigma_\mathrm{z}$. The result can be written in the form of Eq. (11.2-28) as

$$\underline{\mathrm{d}\sigma} = [C_\mathrm{II}]\,\underline{\mathrm{d}\varepsilon}^a - \underline{\mathrm{d}\sigma}_\mathrm{II}^\mathrm{ps} \tag{11.2-33}$$

where for Type II elements

$$[C_\mathrm{II}] = \frac{E}{1-\nu^2}\begin{bmatrix} 1 & 0 & \nu \\ 0 & 0 & 0 \\ \nu & 0 & 1 \end{bmatrix} \tag{11.2-34}$$

and

$$\underline{\mathrm{d}\sigma}_\mathrm{II}^\mathrm{ps} = \frac{E}{1-\nu^2}\begin{bmatrix} 1 & 0 & \nu \\ 0 & 0 & 0 \\ \nu & 0 & 1 \end{bmatrix}\begin{Bmatrix} W_\mathrm{r}\mathrm{d}\bar{\varepsilon}^p + Q_\mathrm{r}\mathrm{dT} + \mathrm{d}\varepsilon^s + \frac{\nu \mathrm{dp_g}}{E} \\ \frac{(1-\nu^2)\mathrm{dp_g}}{E} \\ W_\mathrm{z}\mathrm{d}\bar{\varepsilon}^p + Q_\mathrm{z}\mathrm{dT} + d\varepsilon^s + \frac{\nu \mathrm{dp_g}}{E} \end{Bmatrix} \tag{11.2-35}$$

The singularity of $[C_\mathrm{II}]$ reflects the fact that the radial displacements of the inner and outer radii of a cracked element cannot be determined uniquely from its exterior loads. This is due to the reason that a rigid body radial displacement of an element is a possibility. However, when all of the element stiffness matrices for an axial segment are assembled together, the resulting global stiffness matrix is, in general, non-singular and a unique solution for the displacements can be found.

11.2.2 Algorithm for Separating the Components of the Total Strain

In FPIN2, first the finite element equilibrium equations (described in Section 11.2.4.2) are solved to obtain the increments of total apparent strain. After separating the elastic contribution through the use of algorithm described below, we can then calculate the stresses from Eq. (11.2-11). The algorithm, which is based on a method developed by Schreyer [11-8], also yields the other components of the total strain. This algorithm has been extended in FPIN2 to include crack strains, swelling strains, and more general constitutive laws. The algorithm can be broken into six steps as follows.

Step 1: At the beginning of a time step, t_i, all quantities of interest are assumed to be known. Approximations to the elastic strains at the end of time step, $t_{\mathrm{i}+1}$, are calculated by subtracting the thermal strain (using new temperatures), estimated plastic strain, and the estimated swelling strain from the total strain:

$$\underline{\varepsilon}_\mathrm{N}^e = \underline{\varepsilon}_{\mathrm{i}+1}^t - \underline{\varepsilon}_{\mathrm{i}+1}^\theta - \underline{\varepsilon}_\mathrm{N}^p - \varepsilon_\mathrm{N}^s \tag{11.2-36}$$

Since the algorithm is iterative, an iteration counter, N, is introduced as a subscript. In FPIN2, starting values $(N = 1)$ for plastic strain are estimated using the values at the beginning of the time step and the previous plastic strain rate as follows

$$\underline{\varepsilon}_1^p = \underline{\varepsilon}_\mathrm{i}^p + \frac{\mathrm{d}\underline{\varepsilon}_\mathrm{i}^p}{\mathrm{dt}}\Delta t \tag{11.2-37}$$

Similarly, an initial estimate for the swelling strain is given by

$$\underline{\varepsilon}_1^s = \underline{\varepsilon}_\mathrm{i}^s + \frac{\mathrm{d}\underline{\varepsilon}_\mathrm{i}^s}{\mathrm{dt}}\Delta t \tag{11.2-38}$$

Step 2: The generalized Hooke's law is used to calculate the stresses:

$$\underline{\sigma}_\mathrm{N} = [C]\,\underline{\varepsilon}_\mathrm{N}^e \tag{11.2-39}$$

Step 3: The time rate-of-change of the equivalent plastic strain is calculated from

$$\frac{\mathrm{d}\bar{\varepsilon}^p_{\mathrm{N}}}{\mathrm{dt}} = \frac{\bar{\varepsilon}^p_{\mathrm{N}} - \bar{\varepsilon}^p_{\mathrm{i}}}{\Delta t} \tag{11.2-40}$$

Similarly, the rate-of-change of the swelling strain is calculated from

$$\frac{\mathrm{d}\varepsilon^s_{\mathrm{N}}}{\mathrm{dt}} = \frac{\varepsilon^s_{\mathrm{N}} - \varepsilon^s_{\mathrm{i}}}{\Delta t} \tag{11.2-41}$$

Step 4: Successive evaluations of the plastic flow condition $f_{\mathrm{N}}\left(\sigma_{\mathrm{eN}}, \bar{\varepsilon}^p_{\mathrm{N}}, \mathrm{d}\bar{\varepsilon}^p_{\mathrm{N}}/dt, T_{\mathrm{i}+1}\right)$ and the swelling function $g_{\mathrm{N}}\left(\sigma_{\mathrm{mN}}, \varepsilon^s_{\mathrm{N}}, \mathrm{d}\varepsilon^s_{\mathrm{N}}/\mathrm{dt}, T_{\mathrm{i}+1}, \nabla T_{\mathrm{i}+1}, \rho_{\mathrm{N}}\right)$ are performed using equivalent and mean stresses that are calculated from

$$\sigma_{\mathrm{eN}} = \sqrt{\frac{3}{2}\left(S^2_{\mathrm{rN}} + S^2_{\theta\mathrm{N}} + S^2_{\mathrm{zN}}\right)} \tag{11.2-42}$$

$$\sigma_{\mathrm{mN}} = \frac{\sigma_{\mathrm{rN}} + \sigma_{\theta\mathrm{N}} + \sigma_{\mathrm{zN}}}{3} \tag{11.2-43}$$

Step 5: The increments in equivalent plastic strain and swelling strain are obtained from

$$\Delta\bar{\varepsilon}^p_{\mathrm{N}} = -\frac{f_{\mathrm{N}}}{\frac{\mathrm{df_N}}{\mathrm{d}\bar{\varepsilon}^p}} \tag{11.2-44}$$

$$\Delta\varepsilon^s_{\mathrm{N}} = -\frac{g_{\mathrm{N}}}{\frac{\mathrm{dg_N}}{\mathrm{d}\varepsilon^s}} \tag{11.2-45}$$

These expressions represent the correction term in the Newton-Raphson method that is applied to find the $\bar{\varepsilon}^p$ and ε^s that satisfy the constitutive equations, Eq. (11.2-14) and Eq. (11.2-16). Experience indicates that convergence of the entire algorithm is enhanced if only a single Newton-Raphson correction is performed for each N. Differentiating the plastic flow condition $f = 0$ with respect to $\bar{\varepsilon}^p$ and the swelling function $g = 0$ with respect to ε^s gives

$$\frac{\mathrm{df}}{\mathrm{d}\bar{\varepsilon}^p} = -3G\frac{\partial \mathrm{f}}{\partial \sigma_{\mathrm{e}}} + \frac{\partial \mathrm{f}}{\partial \bar{\varepsilon}_{\mathrm{p}}} + \frac{1}{\Delta t}\frac{\partial \mathrm{f}}{\partial \dot{\varepsilon}_{\mathrm{p}}} \tag{11.2-46}$$

and

$$\frac{\mathrm{dg}}{\mathrm{d}\varepsilon^s} = -3K\frac{\partial \mathrm{g}}{\partial \sigma_{\mathrm{m}}} + \frac{\partial \mathrm{g}}{\partial \varepsilon^s} + \frac{1}{\Delta t}\frac{\partial \mathrm{g}}{\partial \dot{\varepsilon}^s} \tag{11.2-47}$$

where G is the shear modulus and K is the bulk modulus

$$G = \frac{E}{2\left(1+\nu\right)} \tag{11.2-48}$$

$$K = \frac{E}{3\left(1-2\nu\right)} \tag{11.2-49}$$

The total derivatives in Eq. (11.2-46) and Eq. (11.2-47) are evaluated for fixed total strain at $t_{\mathrm{i}+1}$ (the details of these derivations can be found in pp. 23-25 of Ref. 11-2).

Step 6. The last step in the algorithm consists of calculating the new equivalent plastic strain:

$$\bar{\varepsilon}^p_{\mathrm{N}+1} = \bar{\varepsilon}^p_{\mathrm{N}} + \Delta\bar{\varepsilon}^p_{\mathrm{N}} \tag{11.2-50}$$

and the new plastic strain increments by use of the associated flow rule:

$$\underline{\Delta\varepsilon}^p_{\mathrm{N}} = \frac{3}{2\sigma_{\mathrm{eN}}}\Delta\bar{\varepsilon}^p_{\mathrm{N}}\left(\underline{sigma}_{\mathrm{N}} - \underline{sigma}_{\mathrm{mN}}\right) \tag{11.2-51}$$

and from them, the new plastic stains as follows

$$\bar{\varepsilon}^p_{\mathrm{N}+1} = \underline{\varepsilon}^p_{\mathrm{N}} + \underline{\varepsilon}^p_{\mathrm{N}} \tag{11.2-52}$$

Similarly, the new swelling strains are calculated from

$$\bar{\varepsilon}^s_{\mathrm{N}+1} = \underline{\varepsilon}^s_{\mathrm{N}} + \Delta\varepsilon^s_{\mathrm{N}} \tag{11.2-53}$$

It should be emphasized here that the increments in Eq. (11.2-52) and 11.2-53 are added to the plastic and swelling strains corresponding to the last iterate, rather than to the strains corresponding to the beginning of the time step. Iteration continues through steps 1-6 until these increments, which are Newton-Raphson corrections to the strains, are acceptably small.

For a continuous (Type I) element, the total strain is equal to the apparent strain calculated from the equilibrium equations. Therefore, the total strain vector is known at the end of a time step. For Type II elements, existence of radial cracks requires modifications to the algorithm to obtain non-zero crack strain component, ε^e_θ. In Step 1, the component of the elastic strain not obtainable from Eq. (11.2-36), $\varepsilon^e_{\theta\mathrm{N}}$, is calculated by substituting $-p_{\mathrm{g}}$ for $\sigma_{\theta\mathrm{N}}$ in Eq. (11.2-39) and solving this equation for $\varepsilon^e_{\theta\mathrm{N}}$.

When the algorithm is converged, the elastic, plastic, and swelling components of the circumferential and/or axial strains are known for both element types at the end of the time step. Using these values, the total circumferential strain is calculated from

$$\varepsilon^t_{\theta,\mathrm{i}+1} = \varepsilon^e_{\theta,\mathrm{i}+1} + \varepsilon^p_{\theta,\mathrm{i}+1} + \varepsilon^\theta_{\theta,\mathrm{i}+1} + \varepsilon^s_{\theta,\mathrm{i}+1} \tag{11.2-54}$$

And finally, the crack strain is determined from the known "apparent total strain" as follows

$$\varepsilon^c_{\theta,\mathrm{i}+1} = \varepsilon^t_{\theta,\mathrm{i}+1} - \varepsilon^a_{\theta,\mathrm{i}+1} \tag{11.2-55}$$

11.2.3 Continuous Cracking Model

If an element is uncracked in the θ plane, it is reasonable to assume that cracking will occur if the tensile stress σ_θ across that plane exceeds some fracture stress $\sigma_{\theta\mathrm{F}}$. At that point, the element type will change to Type II and a non-zero crack strain ε^c_θ will begin to accumulate. If the crack is closed during the subsequent deformation, the element type will switch back to Type I which does not have cracks. However, consideration of the cracking/closure of the entire element at once may introduce large unphysical perturbations into the mechanical analysis that cause unrealistic results and difficulties with the convergence of the basic solution algorithm. This problem is avoided in FPIN2 by using a continuous cracking model [11-9].

If the tangential stress in the uncracked part of the element is assumed to have reached the fracture stress and the tangential stress across the cracked part of the element equals $-p_{\mathrm{g}}$, the average tangential stress for the element is given by

$$\sigma_\theta = \sigma_{\theta\mathrm{F}} + (\sigma_{\theta\mathrm{F}} + p_{\mathrm{g}})\,\frac{\varepsilon^c_\theta}{\varepsilon^c_0} \tag{11.2-56}$$

where ε^c_0 is a positive constant. The crack penetration distance is assumed to be proportional to the average crack opening strain, ε^c_θ.[5] This continuous cracking equation produces a stress across the fracture surface of a finite element that varies smoothly from the fracture stress down to the gas pressure as the crack strain increases from zero to ε^c_0. Once ε^c_θ exceeds ε^c_0, it is assumed that the stress across the fracture surface remains equal to the gas pressure.

[5] Because of the definition of crack strain in Eq. (11.2-7), ^{c} will be negative for opening of cracks. The crack strain input to and output from FPIN2 is the absolute value of this quantity.

Although a one-dimensional analysis cannot represent the details of the two-dimensional stress field at the tips of actual cracks, it has been shown that the continuous cracking model used in FPIN2 can determine the stable growth of the radial cracks to a reasonable accuracy [11-10]. Incorporation of Eq. (11.2-56) into the mechanical analysis is straightforward, but it leads to somewhat lengthy expressions in relating the stresses to the strains for the cracked elements. The net result changes some of the equations in the algorithm described in Section 11.2.2 and the expressions in the element stiffness matrices as discussed Section 11.2.4.2.

11.2.4 Finite Element Formulation

In this section, the finite element method that is used to derive a set of algebraic equations governing the equilibrium of the fuel and cladding is described. The first step in the derivation is the development of integral forms of the equilibrium equations or, in other words, the virtual work equations.

11.2.4.1 Virtual Work Equations

When all shear stresses are set to zero, the equations of equilibrium in cylindrical coordinates reduce to

$$\sigma_{\mathrm{r,r}} + \frac{\sigma_{\mathrm{r}} - \sigma_{\theta}}{r} = \frac{\sigma_{\theta,\theta}}{r} = \sigma_{\mathrm{z,z}} = 0 \tag{11.2-57}$$

Since the circumferential displacements are either known to be zero or can be determined form algorithm described in Section 11.2.2, developing the virtual work equations for the circumferential displacements is not necessary. By multiplying the equilibrium equation in the r direction by an arbitrary function ϕ_{r}, and equation in the z direction by ϕ_{z}, and integrating the resulting sum over the ring shaped finite element, we get

$$\int_{\mathrm{V}} \left[\left(\sigma_{\mathrm{r,r}} + \frac{\sigma_{\mathrm{r}} - \sigma_{\phi}}{r} \right) \phi_{\mathrm{r}} + \sigma_{\mathrm{z,z}} \phi_{\mathrm{z}} \right] \mathrm{dV} = 0 \tag{11.2-58}$$

Using the formula for the derivative of a product and applying the divergence theorem, Eq. (11.2-58) can be rewritten as

$$\begin{aligned} \int_{\mathrm{V}} \left[\sigma_{\mathrm{r}} \varphi_{\mathrm{r,r}} + \frac{\sigma_{\theta} \varphi_{\mathrm{r}}}{r} + \sigma_{\mathrm{z}} \varphi_{\mathrm{z,z}} \right] \mathrm{dV} &= \int_{\mathrm{V}} \left[\frac{1}{r} \left(r \sigma_{\mathrm{r}} \varphi_{\mathrm{r}} \right)_{\mathrm{r}} + \left(\sigma_{\mathrm{z}} \varphi_{\mathrm{z}} \right)_{\mathrm{z}} \right] \mathrm{dV} \\ &= \int_{\mathrm{S}} \left[\sigma_{\mathrm{r}} \varphi_{\mathrm{r}} \hat{i}_{\mathrm{r}} + \sigma_{\mathrm{z}} \varphi_{\mathrm{z}} \hat{i}_{\mathrm{z}} \right] \cdot \hat{n} \mathrm{ds} \end{aligned} \tag{11.2-59}$$

where $\hat{n}$ is the outward normal to the surface S of volume V, and $\hat{i}_{\mathrm{r}}$, $\hat{i}_{\mathrm{z}}$ are unit vectors in the r and z directions. Assuming that ϕ_{z} has the form

$$\phi_{\mathrm{z}} = \psi z \tag{11.2-60}$$

in which ψ is an arbitrary constant, and also assuming that $\phi_{\mathrm{r}}, \sigma_{\mathrm{r}}, \sigma_{\theta}$, and σ_{z} are all independent of z, Eq. (11.2-59) reduces to

$$\int_{\mathrm{A}} \left[\sigma_{\mathrm{r}} \phi_{\mathrm{r,r}} + \frac{\sigma_{\phi} \phi_{\mathrm{r}}}{r} + \sigma_{\mathrm{z}} \psi \right] \mathrm{dA} = \int_{\mathrm{C}} \sigma_{\mathrm{r}} \phi_{\mathrm{r}} \hat{i}_{\mathrm{r}} \cdot \hat{n} \mathrm{dC} + \psi F \tag{11.2-61}$$

where A is the area of the intersection of the ring shaped finite element with the plane $z = 0$, and C is the curve bounding A. F describes the axial force.

A linear approximation is used for the finite-element displacement function, ϕ_{r}, appearing in the virtual work equations. For the one-dimensional element in Figure 11.2.2, it is given by

$$\phi_{\mathrm{r}} = u_1 N_1 + u_2 N_2 \tag{11.2-62}$$

where u_1 and u_2 are the radial displacements of nodes 1 and 2, and the shape functions are

$$N_1 = \frac{b-r}{l} \tag{11.2-63}$$

and

$$N_2 = \frac{r-a}{l} \tag{11.2-64}$$

Substituting Eq. (11.2-62) into the virtual work equation, Eq. (11.2-61), and integrating over the area of the element, A_e, we get[6]

$$\int_{A_e} [B]^T \, dA = \{t_1, t_2, F\}^T \tag{11.2-65}$$

where $[B]$, t_1 and t_2 are defined as

$$[B] = \frac{1}{1} \begin{bmatrix} -1 & 1 & 0 \\ \left(\frac{b}{r} - 1\right) & \left(1 - \frac{a}{r}\right) & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{11.2-66}$$

$$t_i = \int_C \sigma_r N_i \hat{i}_r \cdot \hat{n} \, dC \quad (i = 1, 2) \ . \tag{11.2-67}$$

11.2.4.2 Incremental Form of the Virtual Work Equations

Eq. (11.2-65) can be viewed as expressing the balance between the external forces on the right hand side and the internal forces on the left hand side. The external forces are known functions of time. The stress field on the other hand must be found to satisfy the Eq. (11.2-65) and the constitutive laws for the material. Since the constitutive equations are, in general, highly non-linear, an incremental technique is used in FPIN2 based on iterations within each increment to calculate the stresses.

Assuming that all the stresses at time t_i are known, the equilibrium equation, Eq. (11.2-65), can be rewritten at the end of the time step, t_{i+1}, as

$$\underline{\Delta f}^{int} = \underline{\Delta f}^{ext} \tag{11.2-68}$$

where

$$\underline{\Delta f}^{int} = \underline{f}^{int}_{i+1} - \underline{f}^{int}_{i} = \int_{A_e} [B]^T \, \underline{\Delta \sigma} \, dA \tag{11.2-69}$$

and

$$\underline{\Delta f}^{ext} = \underline{f}^{ext}_{i+1} - \underline{f}^{ext}_{i} \tag{11.2-70}$$

The problem has now been reduced to finding the stress increments $\underline{\Delta\sigma}$ appearing implicitly in Eq. (11.2-68) through Eq. (11.2-69). These stress increments

$$\underline{\Delta \sigma} = \underline{\sigma}_{i+1} - \underline{\sigma}_{i} \tag{11.2-71}$$

[6] To define ϕ_r over the entire domain of the integration, A, we must cover the domain with a finite element mesh and then sum the contributions at each node from all elements sharing the node. For brevity, this finite element assembly process is not discussed here and formulation is developed for a single element only.

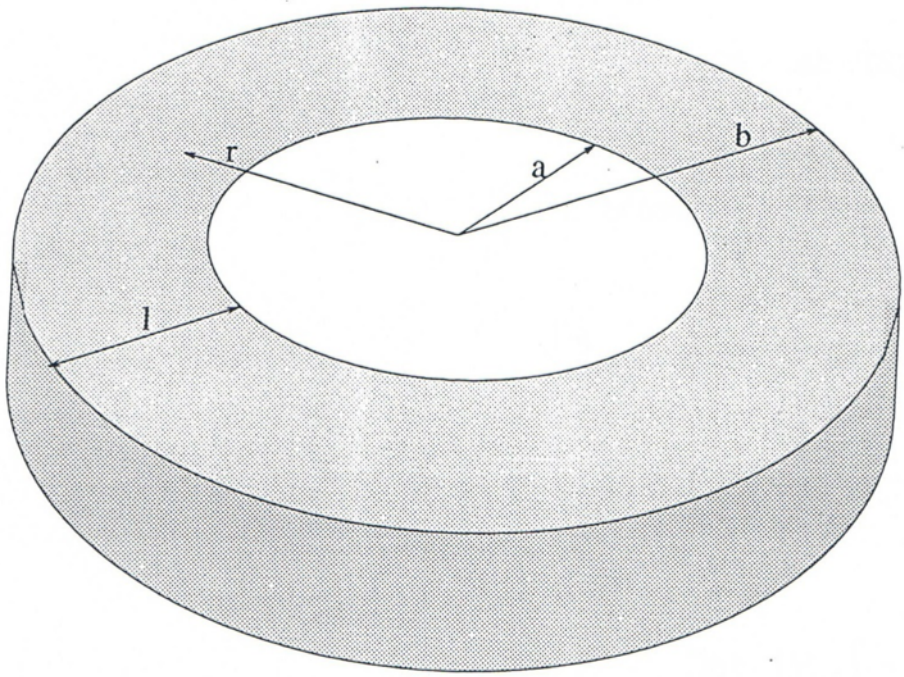


Figure 11.2.2: Finite Element Cross Section

are calculated through an iterative procedure.

As discussed in Section 11.2.1.5, the stress increments are approximated by Eq. (11.2-27) that is repeated here

$$\underline{\Delta\sigma} = [C]_{\mathrm{i}}\, \underline{\Delta\varepsilon}^{a} - \underline{\Delta\sigma}_{\mathrm{i}}^{\mathrm{ps}} \tag{11.2-72}$$

where the elastic modulus, $[C]_{\mathrm{i}}$, and pseudo-stress, $\underline{\Delta\sigma}^{\mathrm{ps}}{}_{\mathrm{i}}$, are evaluated using values at t_{i} (or at $t_{\mathrm{i+1}}$ if known). The apparent strain increment, $\underline{\Delta\varepsilon}^{a}$, in Eq. (11.2-72) can be expressed in terms of displacement increments through the strain-displacement relations, as discussed in Section 11.2.1.1. Using a similar representation with the virtual work functions, ϕ_{r} and ϕ_{z}, radial and axial displacement increments can be written as

$$\Delta u = \Delta u_1 N_1 + \Delta u_2 N_2 \tag{11.2-73}$$

$$\Delta w = z\Delta\psi \tag{11.2-74}$$

Using these in Eq. (11.2-8) leads to

$$\underline{\Delta\varepsilon}^{a} = [B]\, \underline{\Delta\delta} \tag{11.2-75}$$

where

$$\underline{\Delta\delta} = \{\Delta u_1, \Delta u_2, \Delta\psi\}^{T} \tag{11.2-76}$$

After substituting Eq. (11.2-75) into Eq. (11.2-72), and subsequently replacing the result in Eq. (11.2-69), we can rewrite the equilibrium equation, E. Eq. (11.2-68), in the form

$$[K]_{\mathrm{i}}\, \underline{\Delta\delta} = \underline{f}^{\mathrm{ext}} + \underline{f}_{\mathrm{i}}^{\mathrm{ps}} \tag{11.2-77}$$

where

$$[K]_{\mathrm{i}} = \int_{\mathrm{A_e}} [B]_{\mathrm{i}}^{T}\, [C]_{\mathrm{i}}\, [B]_{\mathrm{i}}\, \mathrm{dA} \tag{11.2-78}$$

and

$$\underline{\Delta f}_{\mathrm{i}}^{\mathrm{ps}} = \int_{\mathrm{A_e}} [B]_{\mathrm{i}}^{T}\, \underline{\Delta\sigma}_{\mathrm{i}}^{\mathrm{ps}} \mathrm{dA} \tag{11.2-79}$$

Here $[K]_{\mathrm{i}}$ is the stiffness matrix evaluated at t_{i} using the known values. It should be noted that the forms of the matrices $[C]$ and $[B]$ in Eq. (11.2-78) and $\underline{\Delta\sigma}^{\mathrm{ps}}$ in Eq. (11.2-70) depend on the type of element considered as discussed earlier in Section 11.2.1.5. Explicit formulas for the stiffness matrix and load vectors for both element types considered are given in Section 11.7.1.

11.2.4.3 Iterative Solution of the Equilibrium Equations

Eq. (11.2-77) is the incremental form of the virtual work (or equilibrium) equation. Since the stiffness matrix $[K]_{\mathrm{i}}$ is known, it constitutes a set of linear equations that can be solved for the displacement increments, $\underline{\Delta\delta}$. Eq. (11.2-75) then gives the increment in apparent total strain, $\underline{\Delta\varepsilon}$, and the algorithm described in Section 11.2.2 gives the stresses at time $t_{\mathrm{i+1}}$. These stresses, in turn, can be used in Eq. (11.2-69) to calculate the internal force vector. However, since the expression for the stress increment in Eq. (11.2-72) is only approximate, the equilibrium equation Eq. (11.2-68) will not be satisfied in general.

In FPIN2, another iterative procedure is used to ensure that the stress estimates satisfy the equilibrium equation to an acceptable degree of accuracy. By defining the residual, or the error, in Eq. (11.2-68) as follows

$$\underline{r} = \underline{f}_{\mathrm{i+1}}^{\mathrm{ext}} - \underline{f}_{\mathrm{i+1}}^{\mathrm{int}} \tag{11.2-80}$$

The successive evaluations of the nodal variable increments, $\underline{\Delta\delta}$, are calculated from

$$(K)_{\mathrm{i}}\,\underline{\Delta\delta}_{\mathrm{j}} = \underline{\Delta f}^{\mathrm{ext}} + \underline{\Delta f}^{\mathrm{ps}}_{\mathrm{i}} + \underline{r}_{\mathrm{j}-1} \tag{11.2-81}$$

where j is the iteration index.

The convergence of this iterative scheme is monitored through the norm of the residual vector. If this error is sufficiently small, then the equilibrium equations are assumed to be satisfied and the computation is advanced to the next time step. Otherwise, the new residual vector $\underline{r}_{\mathrm{j}}$, is calculated and added to the right hand side of Eq. (11.2-81) (the previous residual vector, $\underline{r}_{\mathrm{i}+1}$, remains there as well). Figure 11.2.3 summarizes this algorithm for iterative solution of the equilibrium equations.

11.2.5 Large Strain Analysis

Since fuel pin deformation at high temperatures may involve large plastic strains as well as massive local swelling, it is important that the formation of the mechanics model allows for analysis of large strains. Consideration of large strains has impacts on the equilibrium equations derived in Section 11.2.4 and the strain-displacement definitions given in Section 11.2.1. Both changes involve straightforward modifications of the incremental (small perturbation) analysis that has already been derived.

Since the stresses at any time $t_{\mathrm{i}+1}$ must satisfy the equilibrium equations Eq. (11.2-57) in the deformed cylindrical geometry, the virtual work equations should be applied to the fuel element geometry at $t_{\mathrm{i}+1}$ rather than the initial geometry as assumed previously. Following through the derivation of the finite-element equilibrium equation, Eq. (11.2-65), it is easy to show that the only change necessary to accommodate large strains is to redefine the geometric variables to refer to the deformed element. Furthermore, if the incremental deformation between times t_{i} and $t_{\mathrm{i}+1}$ is small, negligible error will be incurred by replacing the geometric variables at $t_{\mathrm{i}+1}$ by these values at t_{i}. These values are readily available form the initial undeformed geometry and the known deformation. From Eq. (11.2-73) and Eq. (11.2-74) for instance, we have

$$a = a_0 + \sum_{\mathrm{i}} \Delta u_1 \tag{11.2-82}$$

$$b = b_0 + \sum_{\mathrm{i}} \Delta u_2 \tag{11.2-83}$$

$$r = r_0 + \sum_{\mathrm{i}} \Delta u \tag{11.2-84}$$

$$z = z_0 + \sum_{\mathrm{i}} \Delta w \tag{11.2-85}$$

where the subscript 0 indicates the initial values, and the sums are taken over the entire deformation history.

The second change in small-perturbation formulation due to large strains is in the strain-displacement definitions. Several large strain measures can be defined that satisfy the necessary requirement of objectivity [11-11]. A convenient large strain measure for the fuel elements is given by the logarithm of the stretches along the principal strain directions. Assuming that the principal strain axes do not rotate for the deformation of interest and the incremental displacements are small, these logarithmic large strain measures can be written as

$$\varepsilon^t_{\mathrm{r,i}+1} = \varepsilon^t_{\mathrm{r,i}} + \frac{\partial}{\partial \mathrm{r_i}}\left(\Delta u\right) \tag{11.2-86}$$

$$\varepsilon^t_{\theta,\mathrm{i}+1} = \varepsilon^t_{\theta,\mathrm{i}} + \frac{\Delta u}{r_{\mathrm{i}}} \tag{11.2-87}$$

$$\varepsilon^t_{\mathrm{z,i}+1} = \varepsilon^t_{\mathrm{z,i}} + \frac{\partial}{\partial \mathrm{z_i}}\left(\Delta w\right) \tag{11.2-88}$$

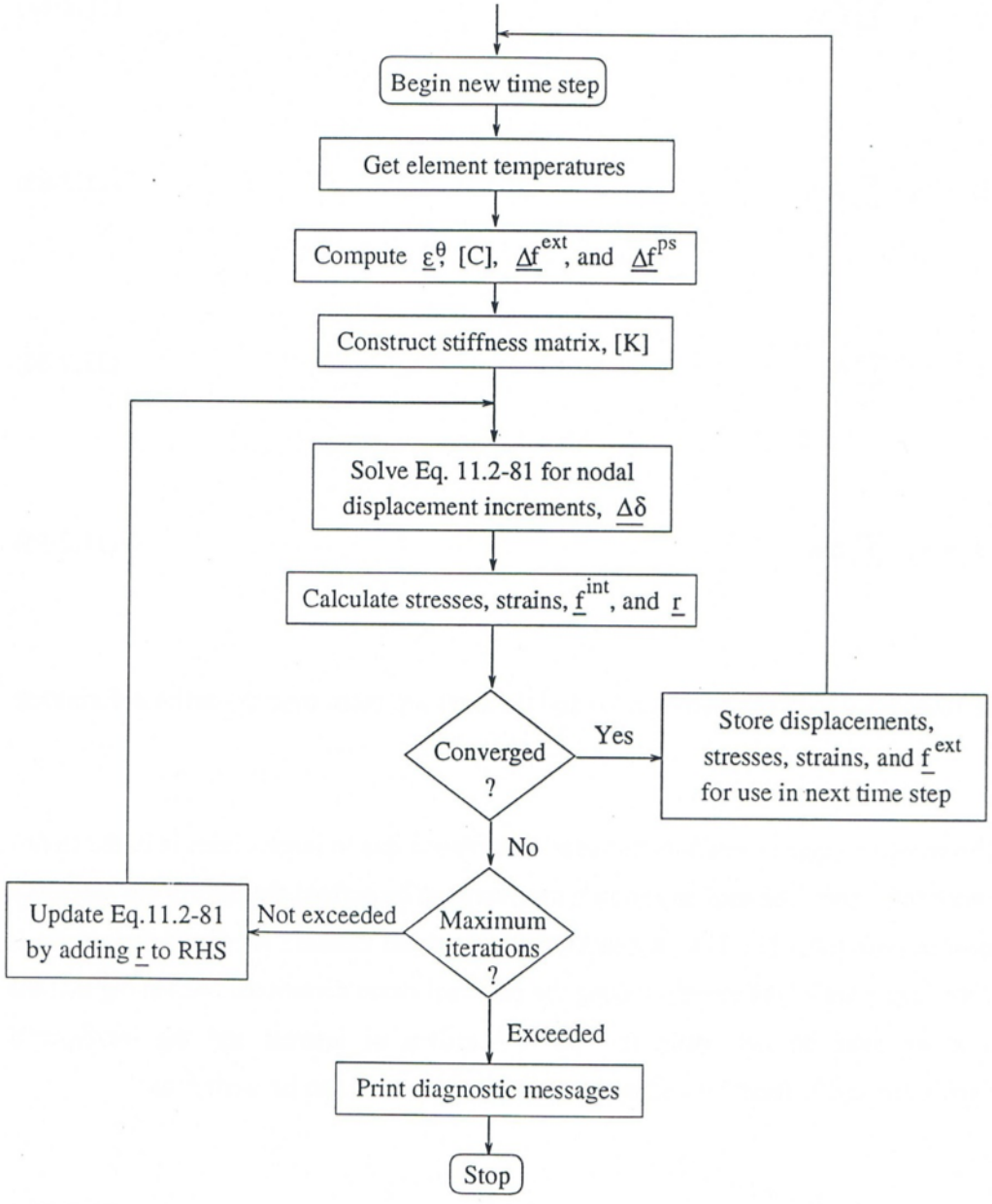


Figure 11.2.3: Iterative Solution of Equilibrium Equations

Comparing Eq. (11.2-86) with Eq. (11.2-5) shows that, in order to use the logarithmic large strain measure, the only change necessary in the analysis given previously is to use the displacement increments to calculate the strain increments and then to determine total strains by adding the increments throughout the deformation history.

11.3 Metal Fuel Element Models

Although the FPIN2 code was originally developed for the purpose of analyzing oxide fuels, it has been modified for the analysis of metallic fuels. The metal fuel version of the code includes numerous improvements to the basic mechanical analysis models that reflect the experience gained from the in-pile TREAT tests on EBR-II irradiated prototypic fuel of the IFR concept. In addition, models for molten cavity formation, the large gas plenum, molten fuel extrusion, and fuel-cladding eutectic formation are provided to complement the fuel element mechanics calculation.

There are many differences between metal and oxide fuels that affect the transient response of the fuel elements. The existence of a large fission gas plenum in metal-alloy fueled elements plays a major role in determining the internal pin pressure during a transient. The sodium bond and comparatively large radial fuel-cladding gap requires consideration of expelling sodium into the plenum as the fuel expands and the gap closes. Metal fuel thermal conductivity is an order of magnitude larger than oxide fuel, and that leads to a flatter radial temperature profile and a more rapid spreading of the region of molten fuel. Since the metal fuel solidus temperature is well below the cladding melting temperature, in many cases fuel can be expected to melt entirely before the cladding fails. Due to the high fuel thermal conductivity, melting usually begins at or near the top of the fuel column with the axial profile of the fuel centerline temperature more closely following the coolant temperature profile than that for oxide fuel. This prevents large cavity pressurization effects in metallic fuels. In addition, a low melting temperature eutectic alloy is expected to form in metallic fuels between the fuel and cladding at temperatures below the anticipated cladding failure. This eutectic formation allows the fuel and cladding to slip freely in the axial direction and can lead to an accelerated cladding failure at elevated temperatures. Cracking is not a significant phenomenon in metallic fuel pins, but metal fuel swelling can be important.

The current version of the FPIN2 coupled with SAS4A/SASSYS-1 includes various model modifications and additions that address these differences between oxide and metal fuels. The following sections contain discussions of these additions and modifications made to FPIN2 for the analysis of metal-fueled elements. The most significant extension to the code is the inclusion of a model for the plenum region above the fuel column as discussed in Section 11.3.1. Extrusion of molten fuel into the plenum as an additional axial expansion mechanism is discussed in Section 11.3.2. Basic mechanical properties of metal-alloy fueled elements are outlined in Section 11.3.3. The constitutive equations for fuel creep and a non-equilibrium approximation to fuel swelling are presented in Section 11.3.4 and Section 11.3.5, respectively. The plastic flow behavior of three types of cladding materials commonly used in metallic fuel elements of the IFR concept is discussed in Section 11.3.6. Fuel-cladding eutectic formation and its impact on the mechanical analysis of fuel elements are summarized in Section 11.3.7. And finally, the fuel element failure formulation is outlined in Section 11.3.8.

11.3.1 Central Cavity and Plenum Models

Internal pin pressure is calculated in FPIN2 using straightforward volume accounting models of the central cavity and the gas plenum. The radial boundary of the cavity for each axial segment is assumed to be at the point where the fuel has reached the solidus temperature. Elements inside the cavity boundary are assumed to be in a hydrostatic state of stress equal to the cavity pressure and are dropped out of stress-strain calculation. These elements occupy a volume in proportion to their density and contribute to the volume and mass balance iteration that determines the cavity pressure.

Metallic fuel elements lead to relatively low transient induced internal pin pressure since (1) fuel porosity easily accommodates the fuel volume expansion upon melting, (2) fuel melting begins at or near the top of fuel column, and (3) metallic fuels have less restrictive flow path for fission gas passage to the plenum. Thus, the large gas plenum is very effective in mitigating any large pressure buildup inside the fuel elements. The pin plenum model of FPIN2 is based on the following assumptions:

1. The internal pin pressure, p_{g}, is considered uniform throughout the interior of a metallic fuel element.
2. The plenum region is assumed to be at the uniform temperature.[1]
3. The thermal expansion of the fill sodium and its expulsion from the fuel-cladding gap as the gap closes is accounted for. However, sodium compressibility is assumed to be negligible at the expected pressure range.

The first level of iteration, or outer loop, in the mechanical analysis section of the FPIN2 is the search for the internal pin pressure p_{g}. According to the calculation sequence, the temperatures of the cavity and plenum materials are known at the beginning of a new time step before the pin pressure is found. Using these temperatures, the following can be calculated:

1. Cavity boundary changes due to fuel melting,

1. Mass of the gas in the cavity, including the gas released upon melting,
2. Mass-average temperature of the gas in the cavity,
3. Volume available from porosity released after fuel melting,
4. The thermal expansion of the sodium in the plenum.

The algorithm for calculating the plenum and cavity pressures are outlined below for the case where the two volumes are connected and share a common pressure, p_{g}. The pin pressure, p_{g}, is calculated by considering the two relationships between the gas volume and pressure that must be satisfied: the ideal gas law, and the interaction between pressure and cavity boundary displacements. Pin molten-cavity and plenum gases are considered as separate entities with individual compositions and temperatures. Since the masses of the gases in the cavity, m_{g}^{C}, and the plenum, m_{g}^{P}, as well as the temperatures of the cavity gas, T_{g}^{C}, and the plenum gas, T_{g}^{P}, are known, the ideal gas law reduces to an inverse relationship between the total gas volume, $V_{\mathrm{g}}^{\mathrm{IGL}}$, and the pressure as follows

$$V_{\mathrm{g}}^{\mathrm{IGL}} = \frac{m_{\mathrm{g}}^{C} R^{C} T_{\mathrm{g}}^{C} + m_{\mathrm{g}}^{P} R^{P} T_{\mathrm{g}}^{P}}{p_{\mathrm{g}}} \tag{11.3-1}$$

where R^{C} and R^{P} are the gas constants for cavity and plenum gases, respectively. The second relationship involves the geometrical calculation of the volume available to the gas that depends on displacement of the solid fuel, cladding, and plenum tube. The liquid fuel and sodium are assumed to be incompressible. Since the gas in the cavity and plenum are considered separately, the total volume has two components

$$V_{\mathrm{g}}^{\mathrm{MECH}}\left(p_{\mathrm{g}}\right) = V_{\mathrm{g}}^{C}\left(p_{\mathrm{g}}\right) + V_{\mathrm{g}}^{P}\left(p_{\mathrm{g}}\right) \tag{11.3-2}$$

where V_{g}^{C} is the available cavity volume that is calculated using the finite element analysis of solid fuel and cladding, and V_{g}^{P} is the volume available for the gas in the plenum. A negative value of V_{g}^{C} means that the molten fuel material is extruded upward into the plenum.

The two values for the total volume, $V_{\mathrm{g}}^{\mathrm{IGL}}$ and $V_{\mathrm{g}}^{\mathrm{MECH}}$, are equal when the correct value of p_{g} is found. In FPIN2, Newton's method is used to fine the value of p_{g} such that

$$f\left(p_{\mathrm{g}}\right) = V_{\mathrm{g}}^{\mathrm{IGL}} - V_{\mathrm{g}}^{\mathrm{MECH}} = 0 \tag{11.3-3}$$

[1] In stand-alone FPIN2 calculation, this plenum temperature is assumed to be equal to the coolant outlet temperature.

The Newton iteration equation

$$p_{\mathrm{g}}^{i+1} = p_{\mathrm{g}}^{i} - \frac{f\left(p_{\mathrm{g}}^{i}\right)}{f'\left(p_{\mathrm{g}}^{i}\right)} \tag{11.3-4}$$

requires the derivative of $V_{\mathrm{g}}^{\mathrm{IGL}}$ and $V_{\mathrm{g}}^{\mathrm{MECH}}$ with respect to p_{g} (i is the iteration counter). The derivative of $V_{\mathrm{g}}^{\mathrm{IGL}}$ is easily calculated from the ideal gas law and the derivative of $V_{\mathrm{g}}^{\mathrm{MECH}}$ is found by forming a finite difference quotient from two calculations of $V_{\mathrm{g}}^{\mathrm{MECH}}$ at two neighboring values of p_{g}.

The total mass of sodium, m_{Na}^{T}, and mass of the gas in the plenum, m_{g}^{P}, are obtained from the initial conditions. The inventory of sodium within the fuel element consists of sodium in the fuel-cladding gap, m_{Na}^{G}, and in the plenum, m_{Na}^{P} :

$$m_{\mathrm{Na}}^{T} = m_{\mathrm{Na}}^{P} + m_{\mathrm{Na}}^{G} \tag{11.3-5}$$

where

$$m_{\mathrm{Na}}^{P} = \rho_{\mathrm{Na}}\left(T^{P}\right)\pi r_{\mathrm{P}}^{2} h_{\mathrm{Na}}^{P} \tag{11.3-6}$$

$$m_{\mathrm{Na}}^{G} = \sum_{\mathrm{j}} \rho_{\mathrm{Na}}\left(T_{\mathrm{j}}\right)\pi\left(r_{\mathrm{ci}}^{2} - r_{\mathrm{fo}}^{2}\right)_{\mathrm{j}} \Delta z_{\mathrm{j}} \tag{11.3-7}$$

The symbols in Eq. (11.3-6) and Eq. (11.3-7) are defined as follows:

ρ_{Na} = sodium density,

T^{P} = plenum temperature,

r_{p} = cladding inner radius in plenum region,

h_{NA}^{P} = height of sodium column in plenum,

r_{ci} = cladding inner radius for axial segment *j*,

r_{fo} = fuel outer radius for axial segment *j*,

Δz_{j} = height of fuel axial segment *j*.

The mass of the plenum gas is determined by using the ideal gas law:

$$m_{\mathrm{g}}^{P} = \frac{P^{P}\pi r_{\mathrm{P}}^{2}\left(h^{P} - h_{\mathrm{Na}}^{P}\right)}{R^{P}T^{P}} \tag{11.3-8}$$

where P^{P} is the plenum pressure and h^{P} is initial plenum height. The plenum pressure is obtained from a user specified reference pressure at a reference temperature as follows

$$P^{P} = P_{\mathrm{ref}}\frac{T^{P}}{T_{\mathrm{ref}}} \tag{11.3-9}$$

During the transient analysis, the cavity pressure algorithm requires the volume in the plenum available to the plenum fission gas and to the material extruded from the molten cavity. This value is calculated by subtracting the volume of the sodium in the plenum from the total volume of the deformed plenum tube. Thus,

$$V_{\mathrm{g}}^{P} = V^{P} - V_{\mathrm{Na}}^{P} \tag{11.3-10}$$

The total plenum volume, V^{P}, is found by assuming that the plenum tube can be treated as a thermoelastic thin shell. The volume of sodium in the plenum is found using the equation

$$V_{\mathrm{Na}}^{P} = \frac{m_{\mathrm{Na}}^{T} - m_{\mathrm{Na}}^{G}}{\rho_{\mathrm{Na}}\left(T^{P}\right)} \tag{11.3-11}$$

11.3.2 Extrusion of Molten Fuel into the Plenum

The procedure in FPIN2 for calculating the behavior of the molten cavity materials uses a simple hydrostatic model based on volume accounting. As discussed in previous section, a volume and mass balance iteration determines the cavity pressure. For transients in which initial fuel melting is observed below the fuel-plenum interface, plenum and cavity pressure equations are solved separately, leading generally to a cavity pressure larger than the plenum pressure. Once melting reaches the top axial segment, the two pressures eventually equilibrate to give a common pin pressure. The temporary imbalance between the cavity and plenum pressures plays an important role in the dynamics of the molten fuel extruded into the plenum.

The extrusion of the molten cavity material into the plenum is calculated at the completion of the pressure iteration and consists of calculating the excess volume of the cavity materials over the volume available in the molten cavity. Both the radial and axial dimensions of the cavity section for each axial segment are based on the deformed geometry. The materials in the cavity include liquid fuel at a temperature above the liquidus temperature, solid fuel at a temperature between the solidus and liquidus, and the fission gas that is assumed to obey the ideal gas law and is released to the cavity in proportion to the melt fraction. Open porosity is available to the fission gas volume when the fuel reaches the solidus temperature, but closed grain boundary porosity is release d in proportion to the metal fraction.

For low-burnup fuel pins, the bond sodium in the fuel-cladding radial gap affects the extrusion of molten fuel into the plenum. The metallic fuel can melt completely across the fuel radial cross section in an axial segment. Although the large gap in fresh metallic fuels is not expected to close by the time of 100% fuel melting, the bond sodium in the gap is expected to escape into the plenum as the molten fuel slumps to fill up the gap due to the large density difference between fuel and sodium. This local slumping lessens the fuel extrusion; however, metal fuels at as low as 0.35 at.% burnup contain sufficient fission gas to result in net upward axial fuel motion [11-12]. Two options are provided in FPIN2 to model this behavior. The first option is to assume that the fuel does not slump and expel the bond sodium. In this case, the fuel outer boundary is assumed to remain at the position it was in at the time of 100% melting. The second (default) option allows the molten fuel to move out to the cladding, with the bond sodium being expelled into the plenum and the gap volume being added to the molten cavity.

To handle the second option, a gap closure model has been added to move the molten fuel out to the cladding over several time steps as the bond sodium is expelled into the plenum and the gap volume becomes available to the molten cavity materials. In order to maintain the stability of the numerical calculation, a varying closure rate is coded in FPIN2 in which the fraction of the gap closed per time step is specified. Somewhat artificial time step dependence of this transition is necessary because the temperature drop across the gap may be as large as 30K. Since the radial temperature profile across the fuel element is relatively flat, an excessively rapid reduction of the gap ΔT may cause computational difficulties such as fuel refreezing during a power rise.

11.3.3 Basic Metal Fuel Properties

Thermal Properties: In the interfaced mode, the SAS-FPIN2 integrated model uses SAS4A/SASSYS-1 routines to calculate fuel and cladding thermal material properties as discussed in Chapter 10. In the stand-alone mode, FPIN2 uses its own built-in correlations for metallic IFR fuel that have been developed by the "Metallic Fuel Properties Working Group" [11-12]. Melting related properties for the metallic fuels such as solidus and liquidus temperatures, heat of fusion, and volume change on melting are calculated using identical algorithms in FPIN 2 and SAS4A/SASSYS-1.

Elastic Properties: Tensile test data at room temperature for uranium and its alloys indicate yielding at very low stresses. The equation used in FPIN2 code for the Young's modulus is as follows

$$E = 0.12 \cdot 10^6 \left(1 - 1.2p\right) \left(1 - 0.754 \cdot 10^{-3} \left(T - 588\right)\right) \tag{11.3-12}$$

where E is Young's modulus (in Bars), p is fractional porosity, and T is temperature (K). This equation gives a value of one-tenth the handbook [11-7] value and should more properly be called a tangent modulus. However, for the expected monotonic loading of most FPIN2 calculations, the fuel is treated as a pseudo-elastic-plastic material with plastic straining handled as secondary creep and yielding is included as part of the "elastic" behavior. In the code, Poisson's ratio is calculated from

$$\nu = 0.27\,(1 - 0.8p)\left(1 + 0.854 \cdot 10^{-3}\,(T - 588)\right) \tag{11.3-13}$$

where ν is Poisson's ratio, p is fractional porosity, and T is temperature (K).

Linear Thermal Expansion: In the metal fuel version of FPIN2, the thermal expansion is expressed in terms of handbook linear thermal expansion data, $\Delta L/L_0$ [11-7]. Using the data rather than the coefficient of thermal expansion, α, results in a more accurate accounting of the total expansion from room temperature to melting since it automatically includes the expansion at solid-to-solid phase transitions. The thermal expansion data and phase transition expansions have been approximated in the code as three straight line segments with breaks at the two solid-to-solid phase change temperature associated with the beginning and ending of the transformation to pure γ phase solid solution. The equations used for the binary fuel are

$$\frac{\Delta L}{L_0} = \begin{cases} 1.695 \cdot 10^{-5}\,(T - 293) & T < 900 \\ 0.0103 + 7 \cdot 10^{-5}\,(T - 900) & 900 < T < 1000 \\ 0.0173 + 2.12 \cdot 10^{-5}\,(T - 1000) & T > 1000 \end{cases} \tag{11.3-14}$$

And, the equations for the ternary fuel are

$$\frac{\Delta L}{L_0} = \begin{cases} 1.67 \cdot 10^{-5}\,(T - 293) & T < 864 \\ 0.0095 + 6.7 \cdot 10^{-5}\,(T - 864) & 864 < T < 950 \\ 0.0153 + 2.12 \cdot 10^{-5}\,(T - 950) & T > 950 \end{cases} \tag{11.3-15}$$

where and T is the temperature (K).

11.3.4 Secondary Creep of Metallic Fuels

Uranium alloys deform plastically under constant load at elevated temperatures. The secondary, or minimum, creep rate is defined as the steady-state rate that is attained under these conditions. The equations added to FPIN2 for metal fuels are obtained from a review of the data and theory for modeling secondary creep of U-Pu-Zr alloys that are of interest to the IFR concept [11-13]. Application of creep data to fuel pin analysis often requires correlating the data using mathematical functions of the governing variables. Such correlations range from purely empirical equations to theoretical equations involving fundamental physical properties. In FPIN2, and intermediate approach is preferred where theoretical models are used to obtain the form of the equation, but the parameters are determined form the creep data rather than the fundamental properties. Such an approach allows an interpolation of the data and gives reasonable confidence in often-required extrapolations beyond the database.

The particular form of the secondary creep equation used in FPIN2 is the form used to represent creep of UO_2 [11-12]. The total plastic strain rate, $\dot{\varepsilon}$, is given by

$$\dot{\varepsilon} = C_1\left(\dot{F}, d\right)\sigma \exp\left(-\frac{Q}{RT}\right) + C_2\left(\dot{F}\right)\sigma^n \exp\left(-\frac{Q}{RT}\right) + C_3\,(T)\,\sigma\dot{F} \tag{11.3-16}$$

where

$\dot{\varepsilon}$ = secondary creep rate, s^{-1}

σ = equivalent stress, MPa

R = universal gas constant, (1.987 cal/g-mole-K)

T = temperature, K

Q = activation energy, (cal/g-mole)

d = grain size, μm

$\dot{F}$ = fission rate, (fissions/cm^3-s)

C = material functions.

In Eq. (11.3-16), the first term represents diffusional creep, the second term dislocation creep, and the third fission-induced creep. Here, the low temperature deformation mechanisms are neglected such as dislocation glide and twinning that may occur in α uranium at temperatures less than 675K. On the other hand, the expected dependence of in-reactor creep on the fission rate is considered.

Some of the parameters in Eq. (11.3-16) are estimated from knowledge of the structure of material and the values that these parameters take for similar materials. At low temperatures, it is reasonable to postulate that the creep deformation of uranium alloys is dominated by the deformation of the α uranium matrix. This is particularly true if the other phases present are coarsely dispersed. When the α uranium matrix deformation is dominant, the activation energy is expected to be the self-diffusion energy in α uranium. The creep activation energy $Q = 52000$ cal/g-mole has been determined experimentally for α uranium. It is shown that the temperature dependence of creep can be deduced using such a single activation energy and that the value of n=4.5 is appropriate in the high stress range [11-13]. Using these values for Q and n, C_1 and C_2 in Eq. (11.3-16) are chosen to minimize the least squares error between the calculated and measures values of $\log(z)$ where z is the Zener-Holloman parameter

$$z = \dot{\varepsilon} e^{\frac{Q}{RT}} \tag{11.3-17}$$

The result is

$$\begin{aligned} C_1 &= 0.5 \cdot 10^4 \quad \text{MPa}^{-1}\text{s}^{-1} \\ C_2 &= 6.0 \quad\quad\quad\ \ \text{MPa}^{-1}\text{s}^{-1} \end{aligned} \tag{11.3-18}$$

When the calculated time to 2% strain using these values is compared to data from creep tests of U-Pu-Zr alloys, results are found to be in acceptable agreement. Examination of experimental data suggests that the grain-size dependence of C_1 can be included as

$$C_1 = \left(\frac{160}{d}\right)^2 0.5 \cdot 10^4 \quad \text{MPa}^{-1}\text{s}^{-1} \tag{11.3-19}$$

for grain sizes near 160 μm.

At temperatures above about 900K, the structure of the U-Pu-Zr alloys no longer contains an α uranium matrix. The transformations between 900K and 975K are complex and depend strongly on composition. Beyond 975K, however, the solid phase consists of a solid solution of uranium, plutonium, and zirconium over a wide range of compositions. In FPIN2, it is assumed that the creep rate in the intermediate temperature range falls between the creep rate of α uranium and solid solution γ phase. It is also assumed that creep of uranium alloys is governed by dislocation glide in the γ regime. The phenomenological creep equation is of the form

$$\dot{\varepsilon} = C_4 \sigma^3 e^{\left(-\frac{Q_\gamma}{RT}\right)} \tag{11.3-20}$$

where Q_γ is the creep activation energy in the γ phase, and the parameter C_4 is, in general, a function of composition and fission rate. Based on tracer diffusion data and/or additional creep data, Q_γ is approximated as the activation energy for creep of the pure γ uranium solvent

$$Q_\gamma = 28500 \quad \text{cal/g-mole} \tag{11.3-21}$$

The constant C_4 is chosen to fit data at 973K as

$$C_4 = 8.0 \cdot 10^{-2} \quad \text{MPa}^{-3}\text{s}^{-1} \tag{11.3-22}$$

Since the data is somewhat limited and the Eq. (11.3-16) and Eq. (11.3-20) are consistent with the creep of alloys with a high percentage of uranium, FPIN2 uses these expressions for all metal fuels.

Experience with earlier versions of the FPIN2 reveals that the evaluation of Eq. (11.3-16) and Eq. (11.3-20) uses a substantial fraction of the computing time. Since the terms involving the exponential function depend only on the temperature, in the current version such terms are evaluated only at the beginning of the mechanics calculation. Therefore, Eq. (11.3-16) is coded in the following form

$$\dot{\varepsilon} = \sigma \left(A + B\sigma^{n-1} \right) \tag{11.3-23}$$

where A and B are previously evaluated constants.

The creep equations given above are the same as those given in the Metallic Fuels Handbook [11-7]. However, fission-induced creep term is not included in the FPIN2 because it does not contribute significantly to the fuel stains on the time scale of the transients that the code has been designed to analyze.

11.3.5 Fission Induced Gas Swelling of Metal Fuels

Irradiated metal fuels experience rapid, large-scale swelling when subjected to overheating. From the safety perspective, this transient swelling could provide an inherent, self-limiting, negative reactivity feedback mechanism. In FPIN2, a transient swelling model has been incorporated to get an estimate for the magnitude of this fission-gas-induced swelling. This non-equilibrium model is based on diffusive growth of grain boundary bubbles.[2]

The distribution of fission gas retained in the fuel matrix is specified as input in FPIN2, and the remainder of the gas is assumed to be in solution or in small bubbles within the fuel grains. Fission gas that is retained in the fuel during steady-state irradiation provides a source for expansion of both solid and liquid fuel during overheating. The amount of gas in the pin plenum is also important since the plenum pressure is a major contributor to cladding loading and, therefore, cladding failure.

The following assumptions are the basis of the transient swelling model of FPIN2 for the analysis of the metallic fuels:

1. All "retained gas" (as measured from experiments on small samples of irradiated fuel) is in solution within the fuel or in the form of small bubbles within the fuel grains or on the grain boundaries.
1. A certain fraction of this gas (input parameter) is contained in grain boundary bubbles, fixed in number and all of the same initial radius (0.1 microns).
2. The open porosity contains fission gas in equilibrium with the plenum pressure.
3. Only the grain boundary bubbles contribute to the swelling of solid fuel.

In order to use this model, a relationship between the swelling strain, ε^s, the mean stress, σ_m, and the fuel temperature, T, is required as explained in Section 11.2.1.2.3. Studies on gas-bubbles growth mechanisms in metallic fuels indicate [11-14] that the transient fuel swelling is dominated by diffusive growth of grain boundary bubbles that is given by

$$\frac{\text{dr}}{\text{dt}} = \frac{\Omega D_{\text{gb}} w \sin^3\theta}{2kTr^2 L\eta}\phi \tag{11.3-24}$$

where

[2] FPIN2 also provides an equilibrium swelling model as an option.

r = bubble radius in the grain boundary,

D_{gb} = grain boundary diffusion coefficient,

w = the boundary thickness,

Ω = the atomic (or molecular) volume, and

kT = thermal energy.

The overpressure, ϕ, in Eq. (11.3-24) is defined as

$$\phi = p_{\text{g}} + \sigma_{\text{m}} - \frac{2\gamma}{\rho} \tag{11.3-25}$$

and it represents the excess of gas pressure, p_{g}, over the sum of the mean stress, σ_{m}, and the surface tension restraint. This restraint is defined in terms of the specific surface free energy, γ, and the radius of curvature, ρ. For the grain-boundary bubbles,

$$\rho = \frac{r}{\sin\theta} \tag{11.3-26}$$

where θ is the intersection angle between the bubble surface and the grain boundary. In FPIN2, the fuel swelling rate is determined from Eq. (11.3-24) for the bubble growth rate and from the assumed bubble density.

The remaining constants in the above equations are geometric constants related to the bubble volume and the bubble spacing [11-14]. The gas in the bubbles is assumed to satisfy the ideal gas law so that

$$p_{\text{g}} = \frac{mRT}{V_{\text{b}}} \tag{11.3-27}$$

where m is the fixed mass of the gas per bubble and V_{b} is the volume of the bubbles. The mass of the gas in a bubble is determined by using the equilibrium conditions at the pretransient time when the bubble radius is assumed to be r_{o}. Thus

$$m = \frac{V_{\text{bo}}}{RT_{\text{o}}}\left(-\sigma_{\text{mo}} + \frac{2\gamma}{r_{\text{o}}}\right) \tag{11.3-28}$$

Since all of the grain boundary gas is assumed to reside in the closed porosity bubbles, the number density of bubbles, N, is given by

$$N = \frac{F_{\text{g}}}{m} \tag{11.3-29}$$

where F_{g} is the grain boundary fission gas density with units of g per cm^3 of fuel. For this simple model, only the swelling of the grain boundary bubbles is considered so that

$$\varepsilon^s = \frac{1}{3}\left(\frac{\Delta V}{V_{\text{o}}}\right) = \frac{1}{3}N\left(V_{\text{b}} - V_{\text{bo}}\right) \tag{11.3-30}$$

These equations, when combined, give the complete specification of the transient swelling model in the form

$$g\left(\varepsilon^s, \sigma_{\text{m}}, T\right) = 0 \tag{11.3-31}$$

The treatment of this constitutive equation for swelling in finite element mechanics is discussed in Section 11.2.2.

11.3.6 Generalized Plastic Flow Behavior of Cladding Materials

The tensile properties of the cladding materials (HT9, D9, and Type 316 stainless steel) are not frequently used in analyses of fuel element performance during normal operation, as designs are usually such that stresses imposed do not challenge the points where yielding should occur. However, under transient conditions at elevated temperatures the applied stresses can result in general yielding. This behavior is modeled in FPIN2 by using a unified plastic deformation model that includes both rate-independent deformation (classical plasticity) and rate-dependent deformation (creep). The equations describing the generalized plastic flow behavior of Type 316 stainless steel, D9, and HT9 cladding are presented in the following sections. The yield strength and ultimate strength of the materials can be calculated from the plastic flow equations by applying them to the geometry of a tensile test.

11.3.6.1 Type 316 SS and D9 Cladding

The deformation behavior of Type 316 stainless steel and D9 cladding can be described over a broad range of temperature and strain-rate with the set of equations shown below. These equations are used in calculations of the strength and deformation at temperatures ranging from room temperature to 1675 K under a variety of loading conditions. In this formalism, the true equivalent flow stress, σ_{e}, is described by the equation

$$\sigma_{\mathrm{e}} = \sigma_{\mathrm{s}} - (\sigma_{\mathrm{s}} - \sigma_{\mathrm{l}}) \exp\left(-\frac{\hat{\varepsilon}}{\varepsilon^*}\right) \tag{11.3-32}$$

where σ_{l} is yield stress of fully annealed unirradiated material, and σ_{s} is the saturation value of the flow stress that is asymptotically approached at large values of plastic strain. Ongoing true plastic strain is incorporated in the "hardness" parameter, $\hat{\varepsilon}$ (input in the model), as this strain is accumulated. The hardness parameter also contains contributions from prior cold work, irradiation hardening, and softening caused by annealing, all scaled as true plastic strain. Consequently, $\hat{\varepsilon} = 0$ for the fully annealed unirradiated material, and $\hat{\varepsilon} = 0.223$ for 20% cold-worked material (default value in the code). Finally, the value of the parameter ε^* is determined by the initial hardening rate θ_{l} of fully annealed material ($\hat{\varepsilon} = 0$) which is given by

$$\theta_{\mathrm{l}} = \frac{\sigma_{\mathrm{s}} - \sigma_{\mathrm{l}}}{\varepsilon^*} \tag{11.3-33}$$

The initial hardening rate, θ_{l}, is found to be only temperature dependent through the shear modulus; so, the rate and temperature dependencies of ε^* is obtained from the above equation.

The yield stress and saturation flow stress are rate and temperature dependent in accordance with the equations

$$\frac{\sigma_{\mathrm{s}}}{G} = \frac{\sigma_{\mathrm{so}}}{G}\left[1 - \exp\left(-\left(\frac{\dot{\varepsilon}_{\mathrm{p}}}{\dot{\varepsilon}_{\mathrm{os}}}\right)^{mK}\right)\right]^{1/K} \tag{11.3-34}$$

$$\frac{\sigma_{\mathrm{l}}}{G} = \frac{\sigma_{\mathrm{lo}}}{G}\left[1 - \exp\left(-\left(\frac{\dot{\varepsilon}_{\mathrm{p}}}{\dot{\varepsilon}_{\mathrm{ol}}}\right)^{mK}\right)\right]^{1/K} \tag{11.3-35}$$

where G is the temperature dependent shear modulus, $\dot{\varepsilon}_{\mathrm{p}}$ is the equivalent plastic strain rate (the $\frac{\mathrm{d}\bar{\varepsilon}^p}{\mathrm{dt}}$ term in Eq. (11.3-26)), m and K are constants, and σ_{so}, σ_{lo}, $\dot{\varepsilon}_{\mathrm{os}}$ and $\dot{\varepsilon}_{\mathrm{ol}}$ are temperature dependent functions. The constant m is the rate sensitivity (the reciprocal of the stress-exponent n in a power-law creep equation), and K is a non-physical fitting parameter that governs the sharpness of the transition between rate-dependent flow and rate-independent flow. At a given temperature, in the high strain-rate limit, the equation for the saturation stress reduces to

$$\frac{\sigma_{\mathrm{s}}}{G} = \frac{\sigma_{\mathrm{so}}}{G} \tag{11.3-36}$$

whereas in the low strain-rate limit, it becomes

$$\frac{\sigma_s}{G} = \frac{\sigma_{so}}{G}\left(\frac{\dot{\varepsilon}_p}{\dot{\varepsilon}_{os}}\right)^m \tag{11.3-37}$$

which reflects the typical power-law creep behavior observed for these materials at high temperatures and low strain rates. The equation for the yield stress behaves similarly. The temperature dependencies of σ_{so} and σ_{lo} are in large part eliminated by dividing by G. However, $\dot{\varepsilon}_{os}$ and $\dot{\varepsilon}_{ol}$ reflect a temperature dependence of high-temperature creep of the form:

$$\dot{\varepsilon}_{os} = \dot{\varepsilon}_{oos} \exp\left(-Q/RT\right) \tag{11.3-38}$$

$$\dot{\varepsilon}_{ol} = \dot{\varepsilon}_{ool} \exp\left(-\frac{Q}{RT}\right) \tag{11.3-39}$$

where $\dot{\varepsilon}_{oos}$ and $\dot{\varepsilon}_{ool}$ are constants; Q, R, and T are the creep activation energy, gas constant, and absolute temperature, respectively. The values for the parameters in all of the above equations are:

$G = 92.0 - 4.0210^{-2}T$ (GPa)

$\frac{\theta_l}{G} = 3.66 \times 10^{-2}$

$\frac{\sigma_{so}}{G} = 2.00 \times 10^{-2} - 9.12 \times 10^{-6}T$

$\frac{\sigma_{lo}}{G} = 2.06times10^{-3} + 7.12 \times 10^{-1}T$

$m = \frac{1}{5.35} = 0.187$

$\frac{Q}{R} = 38533$ (K)

$\dot{\varepsilon}_{oos} = 1.062 \times 10^{14}$ (s^{-1})

$\dot{\varepsilon}_{ool} = 3.794 \times 10^{12}$ (s^{-1})

$K = 2.0$

The combination of the Eq. (11.3-32) and the above strain-rate and temperature laws allow one to generate the complete rate and temperature dependent true-stress/true-strain curves. The yield stress, σ_l, in these equations can be compared to the 0.2% offset yield stress. The ultimate strength, while not specifically denoted in the equations, can be determined by differentiating the Eq. (11.3-32) in accordance with the construction of true stress, σ_u, that corresponds to the engineering ultimate strength:

$$\left.\frac{d\sigma}{d\varepsilon}\right|_{\varepsilon_{pu}} = \theta_l \exp\left(-\frac{\varepsilon_{pu}}{\varepsilon^*}\right) = \sigma_u \tag{11.3-40}$$

where ε_{pu} is the true plastic strain at the ultimate strength which corresponds to the uniform elongation. Solving the above equation and Eq. (11.3-32) simultaneously gives σ_u and ε_{pu}.

11.3.6.2 HT9 Cladding

The deformation behavior of the martensitic-ferritic stainless steel HT9 cladding can be described over a broad range of temperature and strain-rate with the set of equations shown below. These equations are incorporated in FPIN2 and used in calculations of strength and ductility of this alloy at temperatures ranging from room temperature to 1110 K under a variety of loading conditions. The high temperature creep behavior is adequately described by an equation of the Dorn power-law form:

$$\frac{\dot{\varepsilon}_p}{\dot{\varepsilon}_{oos}} = \left(\frac{E}{\sigma_{so}}\right)^n \left(\frac{\sigma_s}{E}\right)^n \exp\left(-\frac{Q_c}{kT}\right) \tag{11.3-41}$$

In this equation, $\dot{\varepsilon}_{\mathrm{p}}$ is the steady-state equivalent creep rate normalized by the constant $\dot{\varepsilon}_{\mathrm{oos}}$, σ_{s} is the equivalent applied stress in the creep test normalized by the constant σ_{so}, E is the temperature dependent Young's modulus, Q_{c} is the creep activation energy, k is Boltzmann's constant, and T is absolute temperature. The temperature dependence of E is expressed as

$$E\left(T\right) = 2.12 \cdot 10^{11}\left[1.144 - 4.856 \cdot 10^{-4} T\right] \quad \text{(Pa)} \tag{11.3-42}$$

and the remaining parameters in the creep equation are given by

$$\dot{\varepsilon}_{\mathrm{oos}} = 5.1966 \cdot 10^{10} \quad \left(\mathrm{s}^{-1}\right)$$

$$\frac{\sigma_{\mathrm{so}}}{E} = 3.956 \cdot 10^{-3}$$

$$n = 2.263$$

$$\frac{Q_{\mathrm{c}}}{k} = 36739 \quad \text{(K)}$$

The flow-stress/strain-hardening behavior can be described by an equation of the type

$$\frac{\sigma_{\mathrm{e}}}{E} = \frac{\sigma_{\mathrm{s}}}{E} - \frac{\sigma_{\mathrm{s}} - \sigma_{\mathrm{l}}}{E} \exp\left(-\frac{\bar{\varepsilon}_{\mathrm{p}}}{\varepsilon^*}\right) \tag{11.3-43}$$

where σ_{e} is the equivalent flow stress at some value of true equivalent plastic strain $\bar{\varepsilon}^p$ (defined in Eq. (11.3-26)), measured from the reference state of as-heat-treated material ($\bar{\varepsilon}^p = 0$). As described in Section 11.3.6.1, θ_{l} is the yield stress of as-heat-treated material, ε^* is a temperature dependent parameter extracted from the hardening rate at the ultimate strength, and σ_{s} is the saturation stress, or steady-state flow stress, approached at a large plastic strain and assumed compatible with the stress in the above steady-state creep equation. It has been found that assuming that the yield stress $\sigma_{\mathrm{l}} = 0.8\sigma_{\mathrm{s}}$ throughout allows the generation of flow stress-strain curves that agree well with data. The true stress at the ultimate strength (maximum load), after differentiating Eq. (11.3-43), is given by

$$\frac{\sigma_{\mathrm{u}}}{E} = \frac{\sigma_{\mathrm{s}} - \sigma_{\mathrm{l}}}{E} \varepsilon^* \exp\left(\frac{-\varepsilon_{\mathrm{pu}}}{\varepsilon^*}\right) \tag{11.3-44}$$

where $\varepsilon_{\mathrm{pu}}$ is the true strain at the ultimate strength corresponding to the uniform elongation. Simultaneously solving this equation in conjunction with the flow stress equation evaluated at $\varepsilon_{\mathrm{pu}}$, the following expression is obtained for the quantity ε^*:

$$\varepsilon^*\left(T\right) = 0.12733 - 3.5027 \cdot 10^{-4} T + 2.9934 \cdot 10^{-7} T^2 \tag{11.3-45}$$

An equation to model the transition between high-rate, low-temperature, rate independent flow behavior and creep behavior has the form

$$\frac{\sigma_{\mathrm{s}}}{E} = \frac{\sigma_{\mathrm{so}}}{E}\left\{1 - exp\left[\left(\frac{\bar{\varepsilon}_{\mathrm{p}} \exp\left(\frac{Q_{\mathrm{t}}}{kT}\right)}{\varepsilon_{\mathrm{oos}}}\right)^{K/n}\right]\right\}^{1/K} \tag{11.3-46}$$

Because of microstructural changes that occur during long-time creep testing of HT9, and the effects they have on flow stress, there is not a smooth transition between short term tensile behavior and creep behavior. However, setting the non-physical fitting parameter as $K = 0.2$, and with all the other parameters set as indicated above, Eq. (11.3-46) follows the tensile data quite well, and reduces to the power-law creep equation at very low strain rates.

For temperatures above the completion of the γ transformation temperature (1233 K), the flow stress model in FPIN2 assumes that the deformation rate for HT9 cladding is the same as that given by the Type 316 SS and D9 cladding equations in Section 11.3.6.1. A simple mixture rule is used to calculate the deformation rates in the α-γ transition region (1110K-1233K).

11.3.7 Fuel-Cladding Eutectic Formation

An additional complication for metallic fuels is the formation of a low melting point eutectic alloy between the fuel and the cladding that can contribute to fuel element failure during transient overheating events. The eutectic alloy forms due to interdiffusion of fuel and cladding constituents at the fuel-cladding interface. It melts at a temperature lower than the fuel and cladding solidus temperatures.

11.3.7.1 Eutectic Penetration of the Cladding

The primary effect of the liquid eutectic alloy formation on fuel element failure is thinning of the cladding wall. The liquid does not further damage the remaining cladding tendon by mechanisms such as liquid metal embrittlement. Experimental data documenting this effect comes from constant temperature time-to-failure experiments on irradiated EBR-II fuel elements and form out-of-pile dipping tests of penetration of iron by molten uranium-iron eutectic alloy. These data have recently been reviewed and the following correlation has been recommended [11-15]

$$\dot{M} = \begin{cases} 0 & T < 1353 \text{ K} \\ 922 + 2.93\,(T-1388) - 0.215\,(T-1388)^2 + 0.001134\,(T-1388)^3 & 1355 \text{ K} \le T \le 1506 \text{ K} \\ \exp\left(22.85 - \dfrac{27624}{T}\right) & T > 1506 \text{ K} \end{cases} \tag{11.3-47}$$

where $\dot{M}$ is the melt rate in μm/s, and T is the absolute temperature in Kelvins. The major feature of the data and these equations is that the melt rate is very rapid for temperatures above 1353 K. For typical cladding dimensions, this means that cladding will completely melt-through is somewhat less than one second once molten fuel reaches 1353K. On the other hand, for the transient rate of concern for a typical FPIN2 application, negligible cladding melt-through occurs until this "rapid eutectic attack" temperature is reached.[3] The Eq. (11.3-47) is used in cladding failure criterion subroutine of FPIN2 to calculate wall thinning.

11.3.7.2 Eutectic Release of Cladding Axial Restraint

When fuel and cladding are in contact, FPIN2 has an option that allows that fuel and cladding to either slip freely axially or to be locked together. In the latter option, it is assumed that the fuel and cladding remain locked, either by metallurgical bonding or by friction, at temperatures below the assumed eutectic alloy melting threshold, T_{th} (an input in the code). At temperatures above T_{th}, the fuel and cladding are assumed to slip freely because of the liquid phase that forms at the fuel cladding interface. This discontinuity in the cladding axial constraint model of FPIN2 at the assumed eutectic alloy melting temperature causes a sudden drop in stress in both fuel and cladding elements as the fuel expands slightly in the axial direction and contracts in the radial direction due to elastic recovery.

When the fuel-cladding gap is open, the fuel and cladding axial displacements are calculated independently. When the fuel and cladding are locked together, the increments in axial strain in the fuel are assumed to be equal to the increments in axial strain in the cladding. This is handled in mechanics calculation by modifying the terms in the stiffness matrix. Specifically, the finite-element algorithm is solved for the resultant fuel and cladding axial forces by combining the fuel and cladding stiffness matrices in a non-iterative procedure. In this procedure, the two equations for the axial displacements are replaced by a single equation for their joint displacement. The redundant equation is then replaced with an identity so that the size of the stiffness matrix and all the associated coding remains the same. When the sudden release of cladding restraint occurs at eutectic alloy melting point, fairly large changes take place in the axial displacements. However, experience has shown that, due to the robust procedure used, this substantial sudden change in the mechanics causes no computational difficulties.

[3] The rate at 1000K is about 10^{-2} μm/s, which would require 10 hours to completely penetrate the full cladding wall thickness.

11.3.8 Fuel Element Failure

The failure of the cladding due to temperature and pressure transients is essentially independent of fuel type except that failure in metal fueled elements is augmented by eutectic formation which penetrates and, in effect, thins the cladding as discussed in the previous section. Also, the loss in cladding stiffness is not considered in calculating the continued deformation of the cladding in the oxide fuel version of FPIN2. For conditions where simultaneous eutectic erosion and plastic deformation are important, oxide fuel version under-predicts the permanent cladding strains at the time of failure. Therefore, the FPIN2 analysis of cladding deformation has been modified to account for this eutectic thinning and to determine the permanent cladding strain remaining after a transient more accurately.

The technique that has been implemented in the FPIN2 cladding deformation model to account for the eutectic attack uses an effective ligament thickness ratio to decease the contribution that the material stresses in a given element make to the overall balance between internal and external forces. Elements that are fully liquid are assumed to have a ligament thickness ratio of zero and those that are solid are assumed to have ligament ratio of one. An element that has partially liquefied then has a ratio equal to the fraction of the element that is still solid. In other words, the effect of eutectic formation is included in FPIN2 by considering only the thickness of unaffected cladding that is available to carry the load. This model has the additional benefit that it can easily be modified to account for other damage mechanisms such as cavitation and void growth during tertiary creep. In the current version of the FPIN2, however, these additional mechanisms are not considered.

For transients with relatively short time scale, the eutectic penetration is a factor only if the cladding temperature exceeds the "rapid eutectic attack" temperature (1353K) where the penetration rate increases by three orders of magnitude. The change in cladding wall thickness above 1353K is calculated using Eq. (11.3-47) for Type 316 SS, D9, and HT9 cladding. At temperatures below 1353K, the rate of eutectic penetration is generally insignificant and assumed to be zero.

Cladding rupture is predicted in the code by using the life fraction criteria. The TCD-2 life fraction criterion [11-16] is used for the fuel elements with D9 and Type 316 stainless steel cladding; and the HEDL transient Dorn parameter correlation [11-17] is used for the fuel elements with HT9-cladding. The fuel adjacency effect in the TDC-2 correlation has been neglected. The life fraction over a time step, dt, is calculated from the rupture time, t_{r}, for the instantaneous average cladding temperature, T_{c}, and hoop stress, σ_{c}. Life fractions are summed in the usual way so that cladding failure is predicted to occur at time t_{f} when

$$\int_0^{t_{\mathrm{f}}} \frac{1}{t_{\mathrm{r}}\left(\sigma_{\mathrm{c}}, T_{\mathrm{c}}\right)} \mathrm{dt} = 1 \tag{11.3-48}$$

The location of the failure is the location of first axial cladding segment to reach a life fraction of 1.0. The hoop stress in the cladding tendon is calculated by the thin-shell equations consistent with the methodology used in developing the life fraction correlations. The thickness of this tendon is reduced by the amount of eutectic attack.

11.4 Integration into SAS4A/SASSYS-1 and Usage of Integrated Model

A general purpose SAS-FPIN2 interface has been designed and most of the communication between the two codes is established at this interface minimizing the impacts of coupling on both codes. This coupling strategy allows for maintaining the stand-alone capability of the two codes while assuring that any future improvements to FPIN2 are automatically reflected in SAS4A/SASSYS-1.

11.4.1 SAS-FPIN2 Coupling Methodology

Two modes of SAS-FPIN2 coupled operations are provided. In the stand-alone mode, FPIN2 reads its own input deck and executes without linking to SAS4A/SASSYS-1. This mode is provided primarily for verification/debugging purposes, to allow independent development of FPIN2, and to allow it to continue its role as a tool for thermo-mechanical analysis of individual fuel pins.

In the interfaced mode, FPIN2 replaces the SAS4A/SASSYS-1 metal fuel element mechanics module DEFORM-5 and calculates the updated dimensions, stresses, and strains at the end of each time step. These FPIN2 results are then made available for use in the analysis of accident energetics by providing estimates of axial expansion of fuel, time and location of cladding failure, and the condition of the fuel element at the time of the failure (although the communication between FPIN2 and the SAS4A in-pin and ex-pin fuel relocation modules, PINACLE and LEVITATE, is not yet established). In the interfaced mode, some of the input necessary to run FPIN2 is included in the SAS4A/SASSYS-1 input deck, and the remaining is interpreted from corresponding SAS4A/SASSYS-1 variables.

11.4.1.1 Stand-alone FPIN2 Calculation

In the stand-alone mode, FPIN2 is run by a call to main FPIN2 driver routine, FPMAIN. In this mode, FPIN2 performs complete thermal-mechanical calculations for a single fuel element. When the heat transfer option to calculate the coolant and structure temperatures is invoked, the fuel element is assumed to be surrounded by a circular coolant channel and an outer wall in pin-in-a-pipe geometry. The code also provides an option in which the cladding outer surface temperature may be specified as a function of time to drop the coolant channel and structure calculations. This mode is mainly provided for direct verification and code debugging purposes.

In the stand-alone mode, FPIN2 input is appended at the end of SAS4A/SASSYS-1 input deck after the END-JOB record. The stand-alone FPIN2 input deck consists of the following records entered in free format:

1. Title,

1. Integer data (including integer debug data),

2. -1/ End of Integers,

3. Decimal data (including decimal debug data),

4. -1/ End of Decimals.

Integer and decimal input data are entered in a form similar to the SAS4A/SASSYS-1 input data. Each of the FPIN2 input variables has an assigned location number. The first entry in a line of input data is the location number of the first member of the set of input data values. The second entry is the number of data values that follow. Integer and decimal input data fields are initialized to zero before reading in the data; therefore, a variable does not have to be read in if it has the value of zero. A line of data may continue over several records; however, the maximum number of data values in a line is limited with 2000. The primary unit system for stand-alone FPIN2 input and output is CGS with exception of pressure which is specified in Bar. Temperature is specified in Kelvin. The full list of FPIN2 input variable for stand-alone calculation along with a brief description for each variable is provided in Section 11.7.2.

11.4.1.2 Interfaced SAS-FPIN2 Calculation

The SAS-FPIN2 interface consists mainly of the steady-state and transient FPIN2 driver routines. The steady-state driver, FPINIT, performs setup of FPIN2 for the interfaced calculation and initializes FPIN2 input from SAS4A/SASSYS-1 data. This routine is executed only once prior to the transient calculations. The transient FPIN2 driver routine, FPDRIV, incorporates the time-advancement scheme and interfaces dynamic variables between the two codes.

During the initialization of FPIN2 from SAS4A/SASSYS-1 input, interface routine FPINIT looks for inconsistencies in fuel element modeling, prints diagnostic messages, and terminates the execution if necessary. The main inconsistencies that a user should be aware of are summarized below:

1. A gas plenum below the fuel column is not allowed in FPIN2; therefore, the execution is terminated if SAS4A/SASSYS-1 input variable **IPLUP** is non-zero.
2. If the fuel element contains axial blankets, i.e., **MZUB** and/or **MZLB** are non-zero, a warning message is printed since FPIN2 mechanics calculation is not normally performed for the axial segments containing blanket fuel.[1]
3. An error message is printed and execution is terminated if the number of axial segments in fuel exceeds FPIN2's limit (20),
4. An error message is printed if SAS4A/SASSYS-1 input variable **IPOWRZ** is non-zero (applicable only when **IHTFLG**≠0).
5. Cross-checking of the input data for abnormalities are also performed (such as zero pin pressure) and diagnostic messages are printed.

Since FPIN2 has been primarily developed for the transient analysis of fuel elements, it lacks models to describe pre-transient irradiation features such as fuel restructuring, fission gas retention and fuel-cladding gap narrowing. These pre-transient conditions are to be provided as input for the metallic fuels as discussed in Section 11.4.3. The as-irradiated geometry, fuel elongation, fission product and porosity distributions, and the effect of fast neutron fluence on cladding are typically obtained from the relevant in-reactor fuel performance database at a desired burnup, or from steady-state fuel performance codes such as LIFE-METAL [11-18] and STARS [11-6]. FPIN2 has the capability to interpret its input from the LIFE-METAL output. This capability is also extended to the integrated SAS-FPIN2 model.

Normally, FPIN2 heat transfer is by-passed in the interfaced mode and the fuel element temperatures are lined with SAS4A/SASSYS-1 calculated fuel, cladding, plenum, and cavity temperatures. To accomplish this by-pass, the FPIN2 mechanics/thermal-hydraulics boundary is identified, and routines that are used in heat transfer calculation are isolated. All the common block variables that are used in the mechanics calculation (but altered in one of these heat-transfer routines) are linked with their SAS4A/SASSYS-1 counterparts. The FPIN2 results for stresses and displacements are in turn made available to SAS4A/SASSYS-1 for the estimates of axial expansion of fuel and associated reactivity effects, time and location of cladding failure, and the condition of the fuel at the time of failure.

In the interfaced mode, setting the input flag, IHTFLG, may also turn on FPIN2's own heat transfer model. The option for including FPIN2 heat transfer model is mainly provided for debugging and code verification purposes. It requires additional data to be interfaced regarding fuel pin heat generation rate and cladding outer surface temperature for each axial segment at each time step as the dynamic boundary condition. When this option is set, FPIN2 uses its own built-in metallic fuel thermal property routines.

In order to establish consistency between SAS4A/SASSYS-1 and FPIN2 calculations in the interfaced mode, some modifications to the FPIN2 code were necessary. Major changes to FPIN2 for this integration are summarized below:

[1] FPIN2 mechanical analysis can be performed for axial blankets by describing them as a type of fuel with known material mechanical properties.

1. Generic precision conversions are performed by combining all type declaration statements in a file and replacing them in each subprogram with an INCLUDE statement referencing this file.
2. Along the same line, type specific intrinsic functions are converted to their generic equivalents.
3. Various table interpolations for time-dependent boundary conditions (pin power and cladding outer surface temperature when IHTFLG=1) are bypassed and these variables are linked with their SAS4A/SASSYS-1 calculated counterparts.
4. Various calls to built-in FPIN2 material thermal property subprograms are also bypassed and the thermal properties that are needed in FPIN2 mechanics calculation are substituted with their SAS4A/SASSYS-1 calculated equivalents.
5. Steady-state and transient pin plenum gas temperature is interfaced with the corresponding SAS4A/SASSYS-1 variable.
6. Constant liquid eutectic alloy melting temperature is converted to a variable and listed as an integrated SAS-FPIN2 model input, XEUTHR.
7. Initial cladding effective inner surface wastage is described as a new integrated SAS-FPIN2 model input variable and incorporated into FPIN2 by defining it as part of the variable for cladding wall thinning due to eutectic penetration.
8. The constant coolant channel pressure is converted to a dynamic array variable and interfaced with its axially varying time-dependent SAS4A/SASSYS-1 counterpart.

All these changes are implemented in such a way that they do not affect the stand-alone performance of FPIN2.

The radial mesh structure for SAS4A/SASSYS-1, stand-alone FPIN2, and interface SAS-FPIN2 calculations are shown in Figure 11.4.1. As presented in Chapter 3, the radial mesh structure in SAS4A/SASSYS-1 for fuel elements at a given axial segment can be set up based on either equal radial difference or equal mass principle. In either case, the boundary nodes are the half sixe as shown in Figure 11.4.1. In FPIN2, on the other hand, finite elements are initially defined in a mesh based on an equal radial difference principle with all elements having the same thickness as shown in Figure 11.4.1. In order to avoid extensive remapping of the thermal and mechanical variables between two meshes during the interfaced calculations, the consistency between the initial mesh structures is accomplished by pulling SAS4A/SASSYS-1-calculated mesh information into FPIN2 common blocks, and forcing FPIN2 to use the same mesh structure in the fuel and cladding (Figure 11.4.1). When FPIN2 heat transfer is by-passed, however, this procedure requires substitution of mesh-centered temperatures for boundary nodes, namely T_{f1}, T_{fNDRF}, T_{c1}, and T_{c3} (locations marked with a "*" in Figure 11.4.1), that are not calculated in SAS4A/SASSYS-1. In the interfaced mode, these temperatures are approximated with a linear interpolation between the temperatures of the neighboring nodes.

Although the fuel element mechanics model of FPIN2 uses an implicit solution scheme, the interaction between the SAS4A/SASSYS-1 thermal-hydraulics and FPIN2 mechanics calculations is explicit. In the SAS4A/SASSYS-1 code system, a multi-level time-step hierarchy is used in which a main time step is divided into one or more primary-loop, heat transfer, and coolant dynamics time steps as described in Chapter 2. The control over the length of a computational time step is performed using a variety of internal and user-specified restrictions. The FPIN2 mechanics calculation is performed at the each heat transfer time step using newly calculated temperatures. Limitations imposed by stability and accuracy requirements assure that heat-transfer time steps are small enough to avoid the problem of unstable results between the thermal and mechanical calculations due to explicit coupling. The implicit treatment in FPIN2 is somewhat inconsistent with the explicit nature of the SAS4A/SASSYS-1 code system and it often results in a notable increase in computation time. However, the capabilities gained by this coupling are often well worth this additional computational cost.

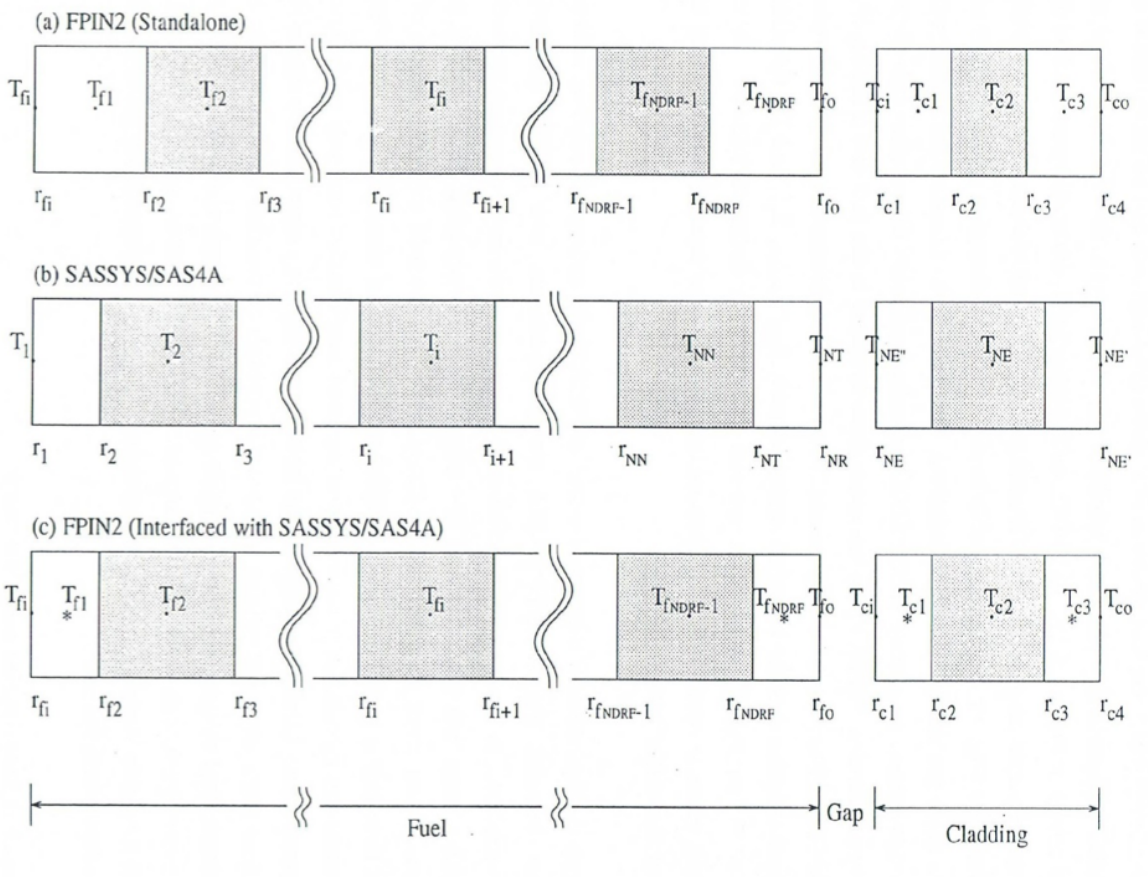


Figure 11.4.1: Radial Mesh Structure and Temperature for an Axial Segment in (a) FPIN2 Code, (b) SAS4A/SASSYS-1, and (c) Integrated SAS-FPIN2 Model.

11.4.1.3 Subroutine Descriptions and Flow-charts

The list of SAS-FPIN2 interface routines is presented in Table 11.4.1. In the stand-alone mode, the FPIN2 main program is called form the SAS4A/SASSYS-1 steady-state driver routine SSTHRM and FPIN2 is executed without interfacing to SAS4A/SASSYS-1. In the interfaced mode, FPIN2 is coupled to the rest of the SAS4A/SASSYS-1 calculations through two main driver subroutines, FPINIT and FPDRIV. First the steady-state FPIN2 driver routine FPINIT is called from SSTHRM for initialization of FPIN2 for interfaced calculations. A flowchart for the FPINIT subroutine is shown in Figure 11.4.2. Then, during the transient calculations, SAS4A/SASSYS-1 thermal-hydraulic manager TSTHRM calls for the transient interface routine FPDRIV that acts as the FPIN2 transient driver.

Table 11.4.1: SAS-FPIN2 Interface Subroutines

Subroutine Name	Description
FPMAIN	In the stand-alone mode, FPIN2 driver (main program)
FPINIT	In the interfaced mode, steady-state FPIN2 initialization routine
FPDRIV	In the interfaced mode, FPIN2 transient driver routine
SASTMP	In the interfaced mode, SAS-FPIN2 thermal-hydraulics interface (when FPIN2 heat transfer module is by-passed)
FPNOUT	In the interfaced mode, output of the FPIN2 results

Subroutine FPDRIV basically includes the section of FPMAIN that is used for the time advancement scheme. This time advancement scheme is preceded by an upper interface routine, SASTMP that pulls SAS4A/SASSYS-1 calculated temperatures into FPIN2 common blocks, and is followed by a lower interface section that takes FPIN2 updated axial and radial mesh information and puts it back into SAS4A/SASSYS-1 common locations. SASTMP is also used by steady-state interface routine FPINIT to extract the initial temperature distribution from SAS4A/SASSYS-1. A flowchart for the subroutine FPDRIV is presented in Figure 11.4.3.

11.4.2 Input Description for Interfaced SAS-FPIN2 Calculation

Two groups of input variables are identified for interfaced SAS-FPIN2 calculations. Integrated model variables that describe the mode of the interfaced calculation and FPIN2 variables that are not readily provided by SAS4A/SASSYS-1 are included in SAS4A/SASSYS-1 common blocks INPCHN (for integer variables) and PMATCH (for real variables). The list of these "new" variables is provided in Table 11.4.2.

Another category includes input variables that are provided by SAS4A/SASSYS-1 calculations and/or are translated from SAS4A/SASSYS-1 input. The list of these variables along with SAS4A/SASSYS-1 counterparts is presented in Table 11.4.3. The variables in Table 11.4.2 and Table 11.4.3 constitute the full list of input parameters necessary to run FPIN2 in the interfaced mode. Further information on some of the input parameters related to pre-transient fuel element characterization is given in the following section. When interpreting FPIN2 input from SAS4A/SASSYS-1 data, dimensional conversions are necessary since the principal unit systems for SAS4A/SASSYS-1 and FPIN2 and SI and CGS, respectively.

Table 11.4.2: Input Variables for Integrated SAS-FPIN2 Model

BLOCK 51 * INPCHN			
Location	Symbol	Value	Definition/Comments
285	IFPIN2	=0	Do not use FPIN2 metal fuel model

continues on next page

Table 11.4.2 – continued from previous page

BLOCK 51 * INPCHN			
		=1	Use FPIN2
(Note: No other data required when IFPIN2=0)			
286	IFPI01	=0	Use FPIN2 in interfaced mode
		=1	Use FPIN2 in standalone mode
(Note: No other data required when IFPI01=1)			
287	IHTFLG	=0	Bypass FPIN2 heat transfer calculation
		=1	Include FPIN2 heat transfer calculation
288	LHTOPT	=0	Perform heat transfer calculation including coolant and wall
		=1	Perform heat transfer calculation with input value of cladding outer surface temperatures
289	LCRACK	=0	No fuel cracking
		=1	Radial fuel cracks included
290	LFPLAS	=0	Allow creep-plastic strains in fuel
		=1	Suppress creep-plastic strains in fuel
291	LCPLAS	=0	Allow creep-plastic strains in clad
		=1	Suppress creep-plastic strains in clad
292	LFSWEL	=0	Allow swelling-hotpressing strains in fuel
		=1	Suppress swelling-hotpressing strains in fuel
293	LCSWEL	=0	Allow swelling strains in clad
		=1	Suppress swelling strains in clad
294	LLRGST	=0	Large strain analysis
		=1	Small perturbation analysis
295	LFCSLP	=0	Fuel-clad locked when gap is closed
		=1	Independent fuel-clad axial displacement
296	LOUTSW	=0	No detailed printing of results - summary only
		=1	Normal detailed printout under LFREQA, MFREQA, and LFREQB control
297	LFREQA		Initial print frequency, number of time steps between normal detailed printout
298	MFREQA		Total number of time steps under LFREQA control
299	LFREQB		Final print frequency
300	LGRAPH	=0	Do not write graphics file
		=1	Write a graphics data file
301	LDBOUT	=0	Do debug output
		=1	Add debug output to regular LOUTSW=2 output
302	LDBSTP	=0	Program stops when molten cavity freezes
		=1	Ignore this program stop
303	LDBFPL	=0	Use recommended fuel flow stress Eq. (11.3-16)
		=1	Use simple power law fuel creep:$\dot{\varepsilon} = C_0 \sigma_e^{C_1}$ (Note: XFPLC0 and XFPLC1 are required)

continues on next page

Table 11.4.2 – continued from previous page

BLOCK 51 * INPCHN			
304	LDBFDV	=0	Use recommended fuel swelling-hotpressing Eq. (11.3-30)
		=1	Use equilibrium swelling model (ANL-IFR-6 and -23)
		=2	Use simple power law fuel swelling: $\dot{\varepsilon} = C_0 \sigma_m^{C_1}$ (Note: XFDVC0 and XFDVC1 are required)
305	LDBCPL	=0	Use recommended clad flow stress
		=1	Ideal plastic flow for clad: $\sigma_y = C_0 + C_1 \bar{\varepsilon}^p$ (Note: XCIPL0 and XCIPL1 are required)
		=2	Use high-temperature power-law creep
		=3	Use simple power law clad creep: $\dot{\varepsilon} = C_0 \sigma_e^{C_1}$ (Note: XCIPL0 and XCIPL1 are required)
306	LGPRES		(Not currently used)
307	LGAPCL	=0	Use fuel-clad opening/closure model
		=1	Fuel-clad gap always closed
308	LCPROP	=0	Use material property correlations (when IHTFLG=1)
		=1	Use temperature independent material properties (when IHTFLG=1)
309	LSKIPM	=0	Perform mechanical calculations
		=1	Bypass mechanical calculations, heat transfer only (when IHTFLG=1)
310	LGCLOS	=0	Use gap closure routine at 100% fuel melting
		=1	Do not close gap (if open) at 100% fuel melting
311-334	LDBOTA(J)		Axial debug print vector (0=no-print, 1=print)
355-345	LDBOTF(I)		Fuel radial debug print vector (0=no-print, 1=print)
346-348	LDBOTC (IC)		Clad radial debug print vector (0=no-print, 1=print)

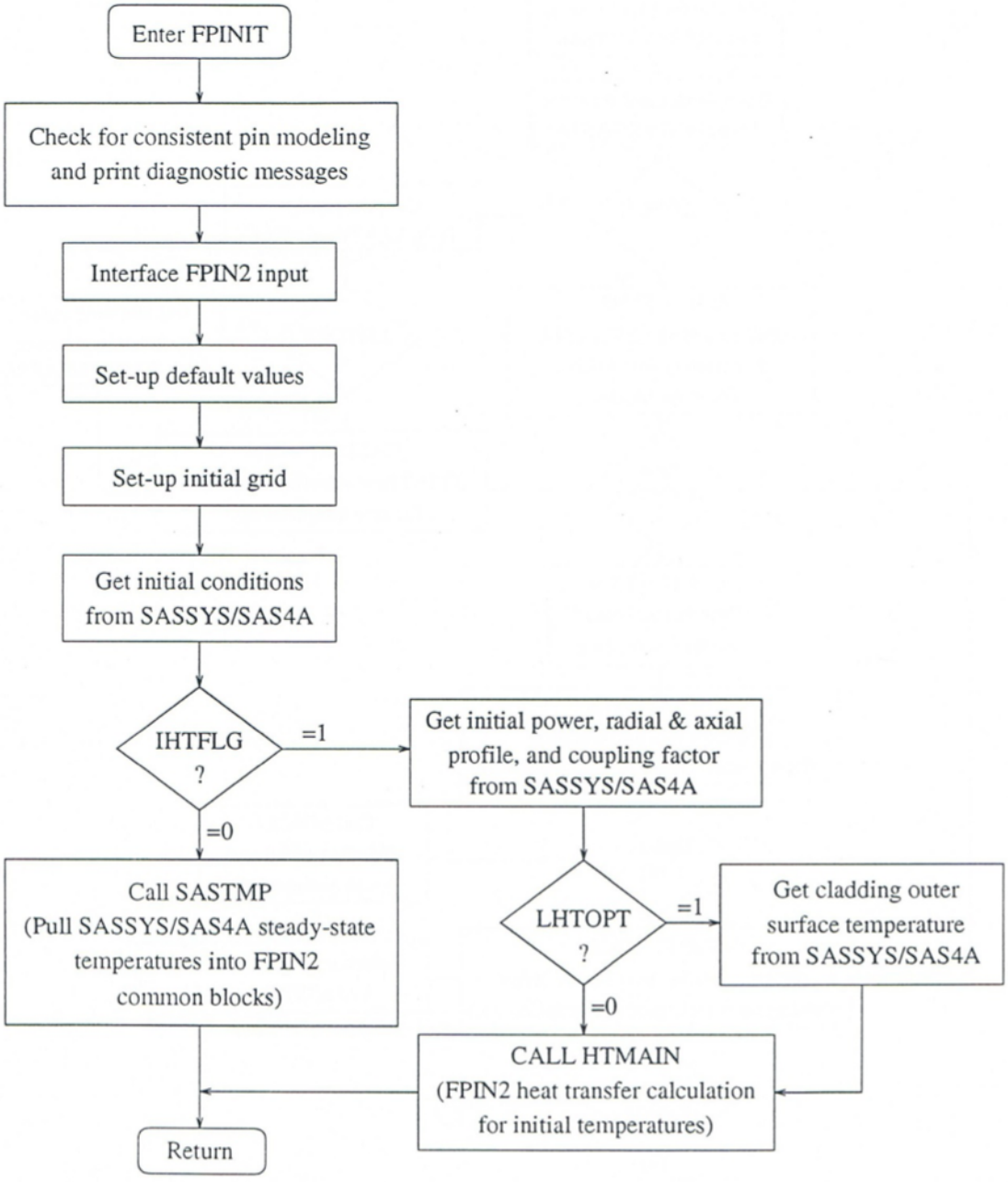


Figure 11.4.2: FPINIT Subroutine Flow Diagram

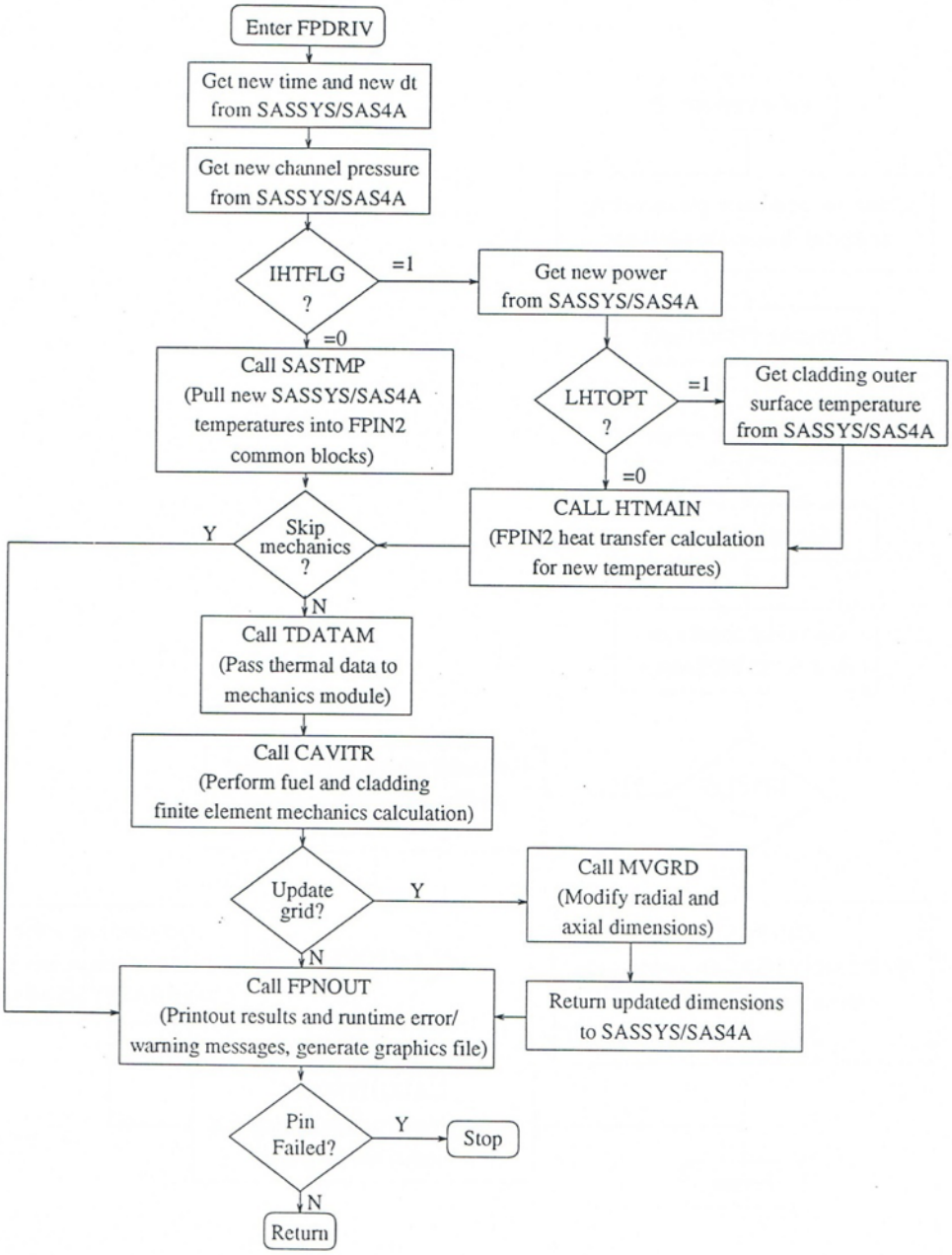


Figure 11.4.3: FPDRIV Subroutine Flow Diagram

BLOCK 63 * PMATCH		
Location	Symbol	Definition/Comments
105	XEUTHR	Liquid eutectic threshold temperature (K) (Default=988.) (See **FSPEC** for consistent input)
106	XGBFRA	(Not currently used)
107-130	XCLDHR(J)	Pre-transient hardness parameter used in clad flow stress calculation. (Default value is 0.223, the value appropriate for 20% CW unirradiated stainless steel.)
131	XFPLC0	Fuel power law creep constant C_0 (when LDBFPL=1)
132	XFDVC1	Fuel power law creep constant C_1 (when LDBFPL=1)
133	XFDVC0	Fuel power law swelling constant C_0 (when LDBFDV=2)
134	XFDVC1	Fuel power law swelling constant C_1 (when LDBFDV=2)
135	XCIPL0	Clad idealized flow stress constant C_0 (when LDBCPL=1 or 3)
136	XCIPL1	Clad idealized flow stress constant C_0 (when LDBCPL=1 or 3)
137	XHTERR	Relative convergence criterion for heat transfer calculation (when IHTFLG=1) (Default=0.0005)
138	XEPSCA	Relative convergence criterion for cavity pressure (Default=0.001)
139	XEPSFE	Relative convergence criterion for finite element analysis (Default=0.0005)
140	XEPTES	Relative convergence criterion for plastic-creep strains (Default=0.0005)
141	XEVTES	Relative convergence criterion for swelling strains (Default=0.0005)

Table 11.4.3: FPIN2 Input Variables that are Provided by SAS4A/SASSYS-1 Calculations and/or Interpreted from SAS4A/SASSYS-1 Input.

Description	FPIN2 Variable	SAS Counterpart
Metal fuel type (U-Fs, binary, or ternary)	IFTYPE	IMETAL
Cladding type (Type 316, D9, or HT9)	ICTYPE	ICTYPE
Number of axial segments in fuel column	NDZ	MZ
Number of radial elements in fuel	NDRF	NT
Number of radial elements in cladding	NDRC	3
Transient initiation time (s)	TZERO	0.
Computation time step size (s)	DTIME	DTP
Initial height of axial segments	DZ(J)	AXHI(J)
Initial length of plenum (cm)	ZPLENM	PLENL
Initial length of bond sodium in plenum (cm)	ZPLNA	BONDNA, pin geometry

continues on next page

Table 11.4.3 – continued from previous page

Description	FPIN2 Variable	SAS Counterpart
Gas constant for plenum gases (Bar-cm^3/gm-K)	PLGASR	RGASSI, HEMM, FGMM, P0GAS, FGFI
Gas constant for central cavity gases (Bar-cm^3/gm-K)	GASCON	RGASSI, HEMM, FGMM
Initial pin pressure (Bar)	PINT	P0GAS
Reference temperature at which PINT is specified (K)	PLTREF	TR
External (channel) pressure (Bar)	PEXT	PCOOL2(J)
Plenum gas temperature (K)	PGASTM	TGASP2
Description	FPIN2 Variable	SAS4A/SASSYS-1 Counterpart
Peak fuel burnup (at.%)	BURNUP	BURNFU
Fuel radial mesh array (cm)	RADF(I,J)	R(I,J)
Clad radial mesh array (cm)	RADC(I,J)	R(NE,J), R(NEP,J)
Distribution of fission gas in closed porosity or in solution (gm/cm^3)	FISGAS(I,J)	ROGSPI, BURNFU, FIFNGB
Fraction of FISGAS on grain boundaries (Default=0.10)	GBFRAC	FIFNGB
Distribution of total fuel porosity	PORES(I,J)	PRSTY2(I,J) (based on PRSTY(IFUELV))
Axial profile of cladding fluence (10^{22} n/cm^2)	CLDFLU(J)	BURNFU, FPDAYS, PBAR(J), FLTPOW, AXHI(J)
Effective cladding inner surface wastage thickness (cm)	WASTE	TWASTI, TWASTO
Mass of fuel elements (gm)	FMASS(I,J)	FUELMS(I,J)
Mass of cladding elements (gm)	CMASS(I,J)	DENSS, cladding geometry.
Distribution of Pu in ternary fuel	FRACPU(I,J)	FUPUMS(I,J), FUELMS(I,J) (based on PUZRTP(IFUELV))
Distribution of Zr in ternary fuel	FRACZR(I,J)	FUZRMS(I,J), FUELMS(I,J) (based on PUZRTP(IFUELV))
Fuel solidus temperature (node-by-node) (K)	FTSOL(I,J)	TSOLIJ(I,J)
Fuel liquidus temperature (node-by-node) (K)	FTLIQ(I,J)	TLIQIJ(I,J)
Normalized time dependent reactor power (when IHTFLG=1)	POWNEW	QMULT
Pin power coupling factor (when IHTFLG=1)	QCONST	1.0
Axial profile of energy generation rate (normalized) (when IHTFLG=1)	QAX(J)	PSHAPE(J)
Radial profile of energy generation rate (W/gm) (when IHTFLG=1)	QR(I,J)	RADPRS(I,J), POW, GAMTNE, GAMTNC, GAMSS, FUELMS
Cladding outer surface temperature (K) (when IHTFLG=1 and IHTOPT=1)	TCSURF(J)	T2(NEP,J)

11.4.3 Pre-transient Fuel Element Characterization

The steady-state fuel element characterization forms the initial conditions from which the transient calculations begin. The FPIN2 mechanics calculation is initiated from a stress-free state for hot and irradiated (swollen) fuel elements. Prior irradiation of fast reactor fuel elements influences their thermal and mechanical response during accident transients significantly. The pre-transient features that are more important for metallic fuels are fuel geometry, fission product, alloy, and porosity distributions, fuel elongation, and the effects of fast neutron fluence on transient cladding properties. These initial conditions can be obtained from the in-reactor fuel performance database or from a fuel performance computer code. In this section, the methods that are suggested to determine the fuel element pre-transient characterization are outlined.

11.4.3.1 Zone Formation

Post-irradiation destructive examinations of ternary fuel pins irradiated in EBR-II reveal significant migration of plutonium and zirconium. This redistribution produces distinct zones that are associated with different metallurgical phases. Micrographs from irradiated ternary fuel pins typically show a three-ring structure that could be separated physically for examination [11-19]. Analysis of these rings reveals that the intermediate zone is depleted in zirconium while the inner and outer zones are enriched in zirconium. The weight fraction of plutonium, on the other hand, remains nearly uniform. The Zr deficient central zone generally consist of the high temperature γ phase in which all constituents are mutually soluble in the solid sate as described in Section 11.3.4. The primary influence of alloy redistribution and zone formation on transient fuel response is through the changes in fuel material properties with alloy content. The most dramatic change is in the fuel solidus temperature where zirconium depletion may lead to initial fuel melting at a radial location other than the hotter axial centerline.

SAS4A/SASSYS-1 provides capabilities to model this fuel composition variation by zone in metallic fuels (**IMETAL**=2) as discussed in Section 10.3.5. This multiple radial fuel zone option is invoked by setting **IFUELC**=1. The distributions are determined by fixing the zone boundaries and the alloy content of the each zone (see input variables **IZNC**, **IZNM**, and **MFTZN**). The representative alloy distributions that are chosen to characterize ternary fuel to be analyzed are specified in the input as weight fractions for each fuel type (**PUZRTP**). These distributions are used in the fuel material property routines to determine variations in properties with alloy content.

SAS4A/SASSYS-1 converts this zone information to internal arrays that describe metal fuel composition and composition-dependent quantities on a node-by-node basis. In the integrated SAS-FPIN2 model, FPIN2 interfaces with these internal SAS4A/SASSYS-1 arrays to initialize variable FRACPU(I,J), FRACZR(I,J), FTSOL(I,J), FTLIQ(I,J) and FMASS(I,J).

11.4.3.2 Fission Gas Distribution

Fission gas plays an important role in transient fuel element response. The gas that is retained in the fuel during steady-state irradiation provides a source for expansion of both solid and liquid fuel during overheating. The quantity of fission gas in the plenum is also important since the plenum pressure is a major contributor to cladding loading. The distribution of fission gas retained in the fuel matrix is specified as input in FPIN2 (FISGAS(I,J)). Part of this gas is assigned to grain boundary bubbles (GBFRAC) and the remainder of the gas is assumed to be in solution or in small bubbles within the fuel grains. A number of models that address the various aspects of fission gas behavior are available. One of these, the STARS code gives a detailed self-consistent picture of the distribution of the gas between the fuel matrix, grain boundaries, edge tunnels, large pores, and the plenum [11-6].

The fraction of the retained gas on the grain boundaries increases with burnup as more gas is released and as the plenum pressure becomes significant compared to surface tension constraint on the grain boundary bubbles.

In the integrated model, the distribution of fission gas in fuel closed porosity and in solution is calculated from SAS4A/SASSYS-1 input variables according to following formula

$$\text{FISGAS (I,J)} = \text{ROGSPI} \cdot \text{BURNFU} \cdot (1 - \text{FIFNGB}) \cdot 1 \cdot 10^{-3} \tag{11.4-1}$$

where the fraction of fission gas on grain boundaries is simply

$$\text{GNFRAC} = \text{FIFNGB} \tag{11.4-2}$$

The unit of FISGAS(I,J) in Eq. (11.4-1) is gm/cm^3. The default value of GBFRAC is 0.10.

The two other relevant input information required by FPIN2 are the plenum and cavity gas constants, PLGASR and GASCON, in Bar-cm^3/gm-K. For fresh fuel pins (BURNFU=0) these constants are simply calculated from

$$\text{GASCON} = \text{PLGASR} = \frac{\text{RGASSI}}{\text{HEMM}} 1 \cdot 10^1 \tag{11.4-3}$$

For irradiated pins (BURNFU≠0) the following formulas are used

$$\text{GASCON} = \frac{\text{RGASSI}}{\text{FGMM}} 1 \cdot 10^1 \tag{11.4-4}$$

$$\text{PLGASR} = \frac{\text{RGASSI} \cdot \text{POGAS} \cdot 1 \cdot 10^1}{\text{POGAS} \cdot \text{FGMM} - (1 - \text{FGFI}) \cdot 1.0133 \cdot 10^5 \cdot (\text{FGMM} - \text{HEMM})} \tag{11.4-5}$$

11.4.3.3 Porosity Distribution

The porosity distribution is also input into FPIN2 (through PORES(I,J) variable). As given in the input description in Table 11.4.3, these values are the total porosity of the nodes, exclusive of the volume of any macroscopic cracks (crack volumes are specified separately as input). The difference between the total porosity and the grain boundary bubble porosity is equal to the porosity of the large pores that are free of surface tension restraint. These large pores may be interconnected (open) or closed. In FPIN2, however, all large pores are assumed to be open and fission gas residing in the open porosity after steady-state irradiation is calculated from the local open pore volume and temperature assuming that the pore pressure is in equilibrium with the plenum pressure. This gas is assumed to be trapped in the fuel during transient heating.

Few measurements of porosity distributions are available for the metallic fuels. Therefore, the fractional porosity is generally determined from the fuel geometry, the fuel mass, and the fuel and fission product densities assuming a uniform distribution. In SAS4A/SASSYS-1, porosity distribution is specified on a zone-by-zone basis using the porosity values for eight fuel types (**PRSTY**) as described in Section 11.4.3.1. Integrated model interfaces with an internal SAS4A/SASSYS-1 array variable (PRSTY2(I,J) to get final distribution on a node-by-node basis.

The magnitude and distribution of the total porosity do not play a large role in mechanics calculation as long as there is sufficient volume to accommodate the grain boundary bubbles so that the resultant open porosity is greater than zero. The FPIN2 models assume that the open porosity does not contribute to solid fuel swelling or to hot pressing because.

1. Sodium logging may partially fill the pores,

1. Driving pressures for swelling are small since the voids are connected to the plenum,

2. Most of the voids are probably large enough so that the time constants for their growth are long compared to the span of the accident transient [11-14].

The distribution of open porosity does not significantly influence the mechanics results either. Although fission gas in the open pores is trapped at the time of fuel melting, its pressure is in equilibrium with the plenum pressure so that this gas contributes little to molten fuel expansion. Most of the expansion comes from the grain boundary gas or the gas in solution in the fuel matrix that has significant swelling potential when it collects into large bubbles following fuel melting. Coalescence of small bubbles into large bubbles is very rapid in liquid fuel [11-20] and is assumed to occur instantaneously in the FPIN2 calcualtion of molten fuel extrusion.

11.4.3.4 Cladding Fluence

The axial distribution of the cladding fluence is needed in life fraction correlation evaluations. In the integrated SAS-FPIN2 model, this variable is determined from the following formula

$$\mathrm{CLDFLU}\,(J) = 8.64 \cdot 10^{-22} \frac{\mathrm{FPDAYS} \cdot \mathrm{FLTPOW} \cdot \mathrm{PBAR}\,(J)}{\mathrm{AXHI}\,(J)} \tag{11.4-6}$$

Were FPDAYS, FLTPOW, and AXHI(J) are SAS4A/SASSYS-1 input variables, and PBAR(J) is an internal SAS4A/SASSYS-1 array variable. The unit of CLDFLU(J) in this equation is 10^{22} neutrons/cm^2. For applications where fast-flux to linear-power ratio (**FLTPOW**) is not available, cladding fluence is approximately set equal to the burnup in at.% (i.e., **BURNFU**).[2]

11.4.3.5 Length of Sodium in Plenum

The initial length of sodium in the pin plenum is expected to be specified for hot and irradiated conditions in FPIN2. In the integrated model, this value is calculated internally using SAS4A/SASSYS-1 subroutine NABOND. This subroutine evaluates the gap thickness for each axial segment and determines the amount sodium in the plenum from initial mass of sodium added to produce the fuel-cladding bound (**BONDNA**). The length in sodium in the plenum then is calculated from plenum geometry consistent with the steady-state temperature distribution.

11.4.3.6 Effective Cladding Inner Surface Wastage

The integrated SAS-FPIN2 model provides an extended capability to include the effects of cladding wastage on fuel element mechanics that is not an original part of the FPIN2 code. The SAS-FPIN2 interface has been modified to include two SAS4A/SASSYS-1 input variables describing initial wastage thicknesses on the cladding inner and outer surfaces (**TWASTI** and **TWASTO**, respectively.)

Cladding wastage is typically considered in design-basis safety assessments. The bases of the wastage in metallic fuel elements are the scratches on the cladding surfaces, diffusion of fuel constituents and fission products into the cladding, and the eutectic formation at the fuel cladding interface. Diffusion of fuel and fission products creates a lanthanide rich FCCI zone that has distinctly different microstructure with cracks and it is assumed to be strengthless. In addition, a separate carbon depleted band with decreased hardness is often identified next to the FCCI zone in HT9 cladding. Although this region exhibits only a moderate decrease in strength, it can conservatively be considered as part of the wastage for the safety cases.

In the integrated SAS-FPIN2 model, effective inner surface wastage is defined as the sum of inner and outer surface wastages and it is incorporated into the mechanics calculation by defining them as part of an FPIN2 variable describing the cladding eutectic penetration. This allows consideration of the wastage bands as part of the cladding for heat transfer while the stress field is determined considering only the thickness of unaffected cladding that is available to carry the load.

[2] EBR-II specific, may not be valid for other reactors.

11.4.4 Output Description and Graphics File Usage

A brief description of the output generated by FPIN2 in the interfaced mode is as follows. A sample of the regular output for fuel/cladding summary information and plenum/molten cavity results is shown in Figure 11.4.4. In the first part of the output, FPIN2 mechanics results are printed for each axial segment separately. The descriptions of the variables printed in this category are presented in Table 11.4.4. FPIN2 regular output for each axial segment is followed by a summary of plenum/molten cavity results as shown in the bottom portion of Figure 11.4.4. The description of the variables printed in this category is presented in Table 11.4.5.

In addition to regular FPIN2 output described above, a series of diagnostic messages are also printed as part of SAS4A/SASSYS-1 output as the integrated model calculations progress. These diagnostic messages can be categorized as follows:

1. Messages regarding the execution of the FPIN2 such as non-convergent iterations and maximum iteration warnings in various parts of the program, and occurrence of non-positive definite matrix,
2. Messages regarding the non-physical phenomena such as negative gas pressure, inconsistent input for constitutive equation options, or negative open porosity,
3. Information messages regarding the cladding failure in a particular axial segment, fuel-cladding gap mixup, complete cladding melting, and cavity solidification stop.

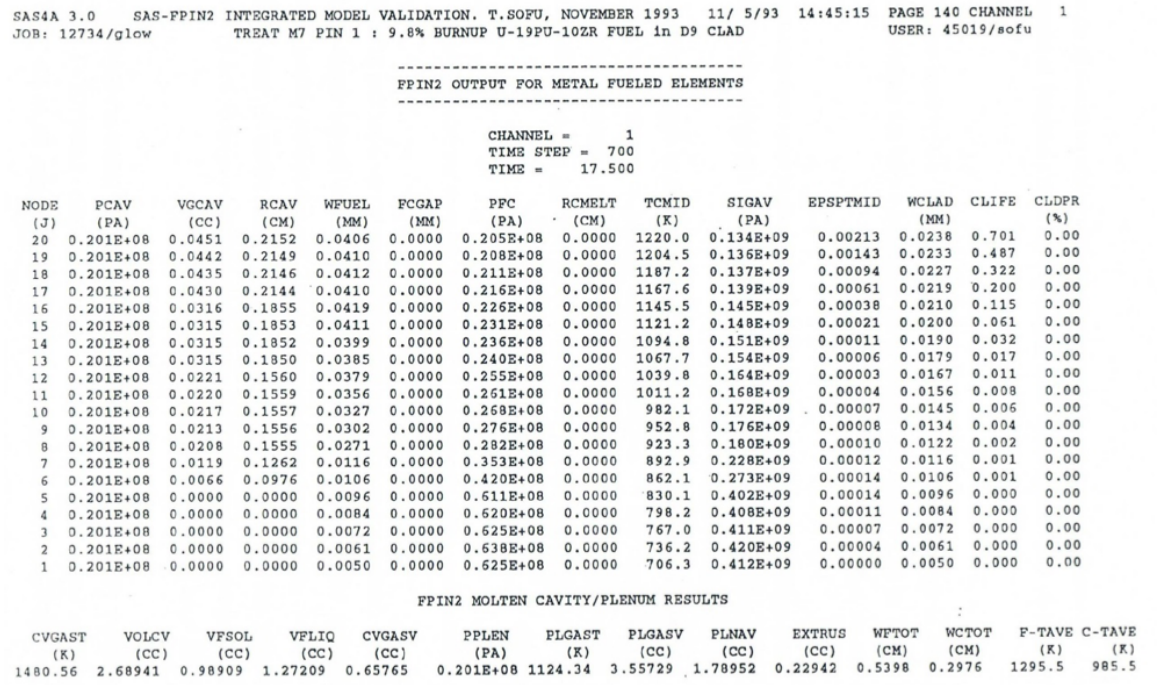
A summary of these messages is also printed at the end of the transient calculations.

The FPIN2 detailed output option can be invoked by setting the input flag **LOUTSW**=1. This option generates a huge printout for the details of FPIN2 calculations and it is used for debugging purposes only. The information printed under this option is generally self-explanatory; therefore, it is not discussed separately here. The print frequency of the regular and detailed FPIN2 output discussed above is controlled by the same SAS4A/SASSYS-1 input parameters that control DEFORM5 output.

For graphics use, some of the variables printed in regular output are stored in binary form in a graphics file at every time step by invoking the option **LGRAPH**=1. The logical unit number assigned for FPIN2 graphics file is 23. The list of variables printed in the graphics file is presented in Table 11.4.6.

Table 11.4.4: Description of the variables in FPIN2 Regular Output for Fuel Element Mechanics Summary

Variable	Unit	Description
PCAV	Pa	Molten fuel cavity pressure
VGCAV	cm^3	Volume of gas in molten fuel cavity
RCAV	cm	Outer radius of the molten fuel cavity
WFUEL	mm	Axial displacement of fuel column segment
FCGAP	mm	Fuel-cladding gap thickness
PFC	Pa	Contact pressure between the fuel and cladding (plenum pressure if gap is open)
RCMELT	cm	Radius of cladding melting
TCMID	K	Temperature of cladding at radial midpoint
SIGAV	Pa	Average hoop stress in cladding
EPSPTMID		Plastic hoop strain in cladding at radial midpoint
WCLAD	mm	Axial displacement of cladding segment
CLIFE		Cladding life fraction
CLDPR	%	Eutectic penetration of the cladding



SAS4A 3.0 SAS-FPIN2 INTEGRATED MODEL VALIDATION. T.SOFU, NOVEMBER 1993 11/ 5/93 14:45:15 PAGE 140 CHANNEL 1
JOB: 12734/glow TREAT M7 PIN 1 : 9.8% BURNUP U-19PU-10ZR FUEL in D9 CLAD USER: 45019/sofu

FPIN2 OUTPUT FOR METAL FUELED ELEMENTS

CHANNEL = 1
TIME STEP = 700
TIME = 17.500

NODE (J)	PCAV (PA)	VGCAV (CC)	RCAV (CM)	WFUEL (MM)	FCGAP (MM)	PFC (PA)	RCMELT (CM)	TCMID (K)	SIGAV (PA)	EPSPTMID	WCLAD (MM)	CLIFE	CLDPR (%)
20	0.201E+08	0.0451	0.2152	0.0406	0.0000	0.205E+08	0.0000	1220.0	0.134E+09	0.00213	0.0238	0.701	0.00
19	0.201E+08	0.0442	0.2149	0.0410	0.0000	0.208E+08	0.0000	1204.5	0.136E+09	0.00143	0.0233	0.487	0.00
18	0.201E+08	0.0435	0.2146	0.0412	0.0000	0.211E+08	0.0000	1187.2	0.137E+09	0.00094	0.0227	0.322	0.00
17	0.201E+08	0.0430	0.2144	0.0410	0.0000	0.216E+08	0.0000	1167.6	0.139E+09	0.00061	0.0219	0.200	0.00
16	0.201E+08	0.0316	0.1855	0.0419	0.0000	0.226E+08	0.0000	1145.5	0.145E+09	0.00038	0.0210	0.115	0.00
15	0.201E+08	0.0315	0.1853	0.0411	0.0000	0.231E+08	0.0000	1121.2	0.148E+09	0.00021	0.0200	0.061	0.00
14	0.201E+08	0.0315	0.1852	0.0399	0.0000	0.236E+08	0.0000	1094.8	0.151E+09	0.00011	0.0190	0.032	0.00
13	0.201E+08	0.0315	0.1850	0.0385	0.0000	0.240E+08	0.0000	1067.7	0.154E+09	0.00006	0.0179	0.017	0.00
12	0.201E+08	0.0221	0.1560	0.0379	0.0000	0.255E+08	0.0000	1039.8	0.164E+09	0.00003	0.0167	0.011	0.00
11	0.201E+08	0.0220	0.1559	0.0356	0.0000	0.261E+08	0.0000	1011.2	0.168E+09	0.00004	0.0156	0.008	0.00
10	0.201E+08	0.0217	0.1557	0.0327	0.0000	0.268E+08	0.0000	982.1	0.172E+09	0.00007	0.0145	0.006	0.00
9	0.201E+08	0.0213	0.1556	0.0302	0.0000	0.276E+08	0.0000	952.8	0.176E+09	0.00008	0.0134	0.004	0.00
8	0.201E+08	0.0208	0.1555	0.0271	0.0000	0.282E+08	0.0000	923.3	0.180E+09	0.00010	0.0122	0.002	0.00
7	0.201E+08	0.0119	0.1262	0.0116	0.0000	0.353E+08	0.0000	892.9	0.228E+09	0.00012	0.0116	0.001	0.00
6	0.201E+08	0.0066	0.0976	0.0106	0.0000	0.420E+08	0.0000	862.1	0.273E+09	0.00014	0.0106	0.001	0.00
5	0.201E+08	0.0000	0.0000	0.0096	0.0000	0.611E+08	0.0000	830.1	0.402E+09	0.00014	0.0096	0.000	0.00
4	0.201E+08	0.0000	0.0000	0.0084	0.0000	0.620E+08	0.0000	798.2	0.408E+09	0.00011	0.0084	0.000	0.00
3	0.201E+08	0.0000	0.0000	0.0072	0.0000	0.625E+08	0.0000	767.0	0.411E+09	0.00007	0.0072	0.000	0.00
2	0.201E+08	0.0000	0.0000	0.0061	0.0000	0.638E+08	0.0000	736.2	0.420E+09	0.00004	0.0061	0.000	0.00
1	0.201E+08	0.0000	0.0000	0.0050	0.0000	0.625E+08	0.0000	706.3	0.412E+09	0.00000	0.0050	0.000	0.00

FPIN2 MOLTEN CAVITY/PLENUM RESULTS

CVGAST (K)	VOLCV (CC)	VFSOL (CC)	VFLIQ (CC)	CVGASV (CC)	PPLEN (PA)	PLGAST (K)	PLGASV (CC)	PLNAV (CC)	EXTRUS (CC)	WFTOT (CM)	WCTOT (CM)	F-TAVE (K)	C-TAVE (K)
1480.56	2.68941	0.98909	1.27209	0.65765	0.201E+08	1124.34	3.55729	1.78952	0.22942	0.5398	0.2976	1295.5	985.5

Figure 11.4.4: FPIN2 Regular Output for Fuel Element Mechanics Summary

Table 11.4.5: Description of the Variable in FPIN2 Regular Output for Plenum/Multen Cavity Results

Variable	Unit	Description
CVGAST	K	Molten fuel average temperature
VOLCV	cm^3	Total molten cavity volume
VFSOL	cm^3	Volume of fuel in molten cavity between solidus and liquidus
VFLIQ	cm^3	Volume of fuel in molten cavity above liquidus
CVGASV	cm^3	Volume of fuel vapor in molten cavity
PPLEN	Pa	Pin plenum pressure
PLGAST	K	Pin plenum average temperature
PLGASV	cm^3	Volume in pin plenum available to gas
PLNAV	cm^3	Volume of sodium in plenum
EXTRUS	cm^3	Volume of molten fuel extruded into the plenum
WFTOT	cm	Total axial displacement of fuel
WCTOT	cm	Total axial displacement of cladding
F-TAVE	K	Average temperature of entire fuel column
C-TAVE	K	Average temperature of cladding tube containing fuel

Table 11.4.6: Description of the Variables Stored in FPIN2 Binary Graphics File (Logical Unit #23)

Order	*Variable*	*Unit*	*Description*
1	TIME	s	Current time at which values of the variables are reported
2	PCAVTY	Pa	Molten fuel cavity pressure
3	CVGAST	K	Molten fuel cavity temperature
4	AMELTF(NDZ)	%	Areal (radial) melt fraction for the top axial segment
5	PPLEN	Pa	Pin plenum pressure
6	PLGAST	K	Pin plenum temperature
7	PFC(NDZ)	Pa	Fuel-cladding contact pressure at top axial segment
8	SIGCM(NDZ)	Pa	Cladding average hoop stress in top axial segment
9	EPSPTM(NDZ)		Cladding average plastic hoop strain in top axial segment
10	FRCPEN(NDZ)	%	Cladding eutectic penetration at top axial segment
11	XLIFEF(NDZ)		Cladding life fraction for top axial segment
12	WCTOT	cm	Total cladding axial displacement
13	WFTOT	cm	Total fuel axial displacement
14	EXTRUL	cm	Length of molten fuel extruded into plenum
15	FUELTL	cm	Total fuel elongation
16	TCLADM(NDZ)	K	Average cladding temperature in top axial segment

11.5 References

11-1. T. H. Hughes, Unpublished information, Argonne National Laboratory, 1977.

11-2. M. Rossow, T. H. Hughes, and J. M. Kramer, Unpublished information, Argonne National Laboratory, 1981.

11-3. T. H. Hughes, Unpublished information, Argonne National Laboratory, 1985.

11-4. T. H. Hughes and J. M. Kramer, "The FPIN2 Code - An Application of the Finite Element Method to the Analysis of the Transient Response of Oxide & Metal Fuel Elements," *Proc. Conf. Sci. & Tech. Fast Reactor Safety*, Guersney, May 1986.

11-5. J. M. Kramer, T. H. Hughes, and E. E. Gruber, Unpublished information, Argonne National Laboratory, 1987.

11-6. J. M. Kramer, T. H. Hughes, and E. E. Gruber, "Validation of Models for the Analysis of the Transient Behavior of Metallic Fast Reactor Fuel," *Proc. Int. Conf. Structural Mechanics in Reactor Technology*, Anaheim, California, vol C, pp. 65-76, August 1989.

11-7. G. L. Hofman, et al., Unpublished information, Argonne National Laboratory, 1985.

11-8. H. L. Schreyer, Unpublished information, Argonne National Laboratory, 1980.

11-9. J. M. Kramer, Unpublished information, Argonne National Laboratory, 1977.

11-10. J. M. Kramer and T. H. Hughes, "Modeling of Fuel Cracking and Fuel Plasticity in LMFBR Fuel Pins During Accident Transients," *Proc. Seminar in Mathematical/Mechanical Modeling of Reactor Fuel Elements*, August 1977.

11-11. A. C. Eringen, Nonlinear Theory of Continuous Media, McGraw-Hill, New York, 1962.

11-12. J. M. Kramer and T. H. Hughes, Unpublished information, Argonne National Laboratory, 1993.

11-13. R. G. Pahl, et al., Unpublished information, Argonne National Laboratory, 1986.

11-14. E. E. Gruber and J. M. Kramer, Unpublished information, Argonne National Laboratory, 1985.

11-15. T. H. Bauer, G. R. Fenske, and J. M. Kramer, "Cladding Failure Margins for Metallic Fuel in the Integral Fast Reactor," *Proc. 9th Int. Conf. Structural Mechanics in Reactor Technology*, Lausanne, vol. C, pp. 31-38, 1987.

11-16. W. F. Brizes and M. L. Hamilton, Unpublished information, Hanford Engineering Development Laboratory, 1982.

11-17. F. R. Shober, Unpublished information, Hanford Engineering Development Laboratory, 1986.

11-18. M. C. Billone, Personal Communication (LIFE-METAL seminar Series), May 11, 1993.

11-19. W. F. Murphy, et al., "Post-irradiation Examination of U-Pu-Zr Fuel Elements Irradiated in EBR-II to 4.5 Atomic Percent Burnup," ANL-7602, Argonne National Laboratory, November 1969.

11-20. J. M. Kramer, Unpublished information, Argonne National Laboratory, 1978.

11.6 Nomenclature

Subscript	Description
l	Fully annealed material
θ	Angular direction in cylindrical coordinates
θ, o	Initial, reference
c	Cladding
ci	Cladding inner radius
e	Equivalent
f	Fuel
f	Failure
fo	Fuel out radius
g	Gas
gb	Grain boundary
i	Time step index
i	Radial element index
I, II	Element types
j	Axial segment index
m	Mean, average
n	Iteration index
N	Iteration index
Na	Sodium
pu	True plastic (strain)
r	Radial direction in cylindrical coordinates
r	Rupture
s	Saturation
th	Threshold
u	True/ultimate (stress)
z	Axial direction in cylindrical coordinates
Superscript	**Description**
θ	Thermal (strain)
a	Apparent (strain)

continues on next page

Table 11.6.1 – continued from previous page

Subscript	Description
c	Crack (strain)
C	Cavity
e	Elastic (strain)
ext	External loads
G	Fuel-cladding gap
i	Iteration counter
int	Internal stress field
IGL	Ideal gas law
$MECH$	Mechanical analysis
p	Plastic (strain)
ps	Pseudo (stress)
P	Plenum
ref	Reference
s	Swelling (strain)
t	Total (strain)
T	Total
Notation	**Description**
¯ (overline)	Equivalent
_ (underline)	Indicates a vector
• (dot)	Indicates derivative with respect to time
ˆ (hat)	Indicates a unit vector
[ ]	Indicates a matrix
$\{\}^T$	Transpose of a vector
∇	Gradient
Symbol	**Description**
α	Thermal expansion coefficient
δ	Nodal displacement
$\Delta L/L_0$	Linear thermal expansion
Δt	Time step size
ε	Strain
$\hat{\varepsilon}$	Cladding hardness parameter
ϕ	Overpressure in grain boundary bubble growth model
ϕ	Finite element displacement function
γ	Specific surface free energy in surface tension restraint term
ν	Poisson's ratio
Ω	Atomic volume in grain boundary bubble growth model
ψ	Constant in finite element displacement function in axial direction
ρ	Radius of curvature in surface tension restraint term
ρ	Density
σ	Stress
θ	Intersection angle between the bubble surface and grain boundary
θ	Initial hardening rate
a	Finite element inner radius
A	Area of intersection between the finite elements
b	Finite element outer radius
B	Matrix in equilibrium equation
C	Curve bounding A
C	Matrix in generalized Hooke's law
C	Material functions in secondary creep equation

continues on next page

Table 11.6.1 – continued from previous page

Subscript	Description
$d\varepsilon$	Incremental strain
$d\lambda$	Proportionality factor for plastic strain increments
$d\sigma$	Incremental stress
dp	Pressure increment
D	Grain size
D	Diffusion coefficient in grain boundary bubble growth model
E	Young's modulus
F	Constitutive function for plastic strains
F	Force
F	Axial force
$\dot{F}$	Fission rate
G	Constitutive function for swelling strains
G	Shear modulus
$\hat{i}$	Unit vector in coordinate directions
K	Stiffness matrix
K	Bulk modulus
L	Finite element radial thickness
M	Rate sensitivity parameter in cladding yield and saturation flow stresses
M	Mass
$\dot{M}$	Eutectic penetration rate
$\hat{n}$	Outward normal to the finite element surface, S
N	Finite element shape function
N	Number density of gas bubbles in grain boundary bubble growth model
P	Pressure
Q	Activation energy in secondary creep equation
Q	Thermal modulus
R	Residual in iterative solution of equilibrium equations
R	Radius
R	Gas constant
S	Deviatoric stress
S	Surface area of finite element
T	Time
T	Elements of load vector in equilibrium equation expressing external forces
T	Temperature
U	Displacement in radial coordinate direction
V	Displacement in θ coordinate direction
V	Volume of finite element
W	Grain boundary thickness
W	Displacement in axial coordinate direction
W	Gradient vector for plastic strain increments
Z	Zener-Holloman parameter

11.7 Appendices

11.7.1 Explicit Formulas for Stiffness Matrix and Load Vector

The stiffness matrix is defined in Section 11.2.4.2, Eq. (11.2-78).

$$[K] = \int_{A_c} [B]^T [C] [B] \, \mathrm{dA} \tag{11.7-1}$$

where the form of [B] and [C] matrices depends on the type of fuel element considered. Introducing new variables

$$d = \frac{b}{r} - 1 \tag{11.7-2}$$

$$e = 1 - \frac{a}{r} \tag{11.7-3}$$

the matrix [B] can be written in a compact form as

$$[B] = \frac{1}{l} \begin{bmatrix} -1 & 1 & 0 \\ d & e & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{11.7-4}$$

where a, b, and l are explained in Figure 11.2.2. As noted in Section 11.2.4.1, this form of matrix [B] is valid for both continuous and cracked elements.

Expanding the matrix triple product in Eq. (11.7-1) and using the symmetry of [C] yields

$$[K] = \frac{1}{l^2} \int_{A_c} \begin{bmatrix} \left(C_{11} - 2C_{12}d + C_{22}d^2\right) & \left(-C_{11} + C_{12}\left(d - e\right) + C_{22}\mathrm{ed}\right) & \left(-C_{13} + C_{23}d\right) l \\ \left(-C_{11} + C_{12}\left(d - e\right) + C_{22}\mathrm{ed}\right) & \left(C_{11} - 2C_{12}e + C_{22}e^2\right) & \left(-C_{13} + C_{23}e\right) l \\ \left(-C_{13} + C_{23}d\right) l & \left(C_{13} + C_{23}e\right) l & C_{33} l^2 \end{bmatrix} \mathrm{dA} \tag{11.7-5}$$

Inspection of this equation indicates that all elements of $[K]$ can be expressed in terms of following six integrals

$$I \equiv \int_{A_c} \mathrm{dA} = \pi l \left(a + b\right) \tag{11.7-6}$$

$$I_{\mathrm{d}} \equiv \int_{A_c} \left(d\right) \mathrm{dA} = \pi l^2 \tag{11.7-7}$$

$$I_{\mathrm{d2}} \equiv \int_{A_c} \left(d^2\right) \mathrm{dA} = \pi l \left[\frac{2b^2}{l} \ln\left(\frac{b}{a}\right) - 3b + a\right] \tag{11.7-8}$$

$$I_{\mathrm{e}} \equiv \int_{A_c} \left(e\right) \mathrm{dA} = \pi l^2 \tag{11.7-9}$$

$$I_{\mathrm{e2}} \equiv \int_{A_c} \left(e^2\right) \mathrm{dA} = \pi l \left[b - 3a + \frac{2a^2}{l} \ln\left(\frac{b}{a}\right)\right] \tag{11.7-10}$$

$$I_{\mathrm{ed}} \equiv \int_{A_c} \left(\mathrm{ed}\right) \mathrm{dA} = \pi l \left[a + b - \frac{2ab}{l} \ln\left(\frac{b}{a}\right)\right] \tag{11.7-11}$$

The domain of these integrations is shown in Figure 11.2.2. In terms these newly defined variables, the stiffness matrix can be rewritten as

$$[K] = \frac{1}{l^2} \begin{bmatrix} \left(C_{11}I - 2C_{12}I_{\mathrm{d}} + C_{22}I_{\mathrm{d2}}\right) & \left(-C_{11}I + C_{12}\left(I_{\mathrm{d}} - I_{\mathrm{e}}\right) + C_{22}I_{\mathrm{ed}}\right) & \left(-C_{13}I + C_{23}I_{\mathrm{d}}\right) l \\ \left(-C_{11}I + C_{12}\left(I_{\mathrm{d}} - I_{\mathrm{e}}\right) + C_{22}I_{\mathrm{ed}}\right) & \left(C_{11}I - 2C_{12}I_{\mathrm{e}} + C_{22}I_{\mathrm{e2}}\right) & \left(C_{13}I + C_{23}I_{\mathrm{e}}\right) l \\ \left(-C_{13}I + C_{23}I_{\mathrm{d}}\right) l & \left(C_{13}I + C_{23}I_{\mathrm{e}}\right) l & C_{33} I \, l^2 \end{bmatrix} \tag{11.7-12}$$

The load vector for a pressure load acting in the radial direction in given by Eq. (11.2-67)

$$t_{\mathrm{i}} = \int_{\mathrm{C}} \sigma_{\mathrm{r}} N_{\mathrm{i}} \hat{i}_{\mathrm{r}} \cdot \hat{n} \mathrm{dC} \tag{11.7-13}$$

(i=1,2). The shape functions N_1 and N_2 are defined in Eq. (11.2-63) and Eq. (11.2-64), respectively. For the case of an element bordering the central cavity, the curve C corresponds to the inner radius of the element, we have

$$\sigma_{\mathrm{r}} = -p_{\mathrm{cav}} \tag{11.7-14}$$

$$\hat{i}_{\mathrm{r}} \cdot \hat{n} = \hat{i}_{\mathrm{r}} \cdot \left(-\hat{i}_{\mathrm{r}}\right) = -1 \tag{11.7-15}$$

$$N_1 = 1 \tag{11.7-16}$$

$$N_2 = 0 \tag{11.7-17}$$

$$\int_{\mathrm{C}} \mathrm{dC} = 2\pi r_{\mathrm{cav}} \tag{11.7-18}$$

Hence,

$$t_1 = 2\pi p_{\mathrm{cav}} r_{\mathrm{cav}} \tag{11.7-19}$$

$$t_2 = 0 \tag{11.7-20}$$

A similar derivation for an element experiencing a pressure p_{out} directed radially inward shows that

$$t_1 = 0 \tag{11.7-21}$$

$$t_2 = 2\pi p_{\mathrm{cav}} r_{\mathrm{out}} \tag{11.7-22}$$

where r_{out} is the outer radius of the element.

11.7.2 List of Input Variables for Standalone FPIN2 Calculation

*** Location Numbers for Integer Data ***

(Number in parentheses following value name states maximum value)

Table 11.7.1: COMMON /INPUTI/

Location Number	Name	Value	Description
1	IFTYPE	=1	Uranium - 5% fissium fuel
		=2	Uranium - 10% zirconium fuel
		=3	Uranium - 15% plutonium - 10% zirconium fuel
		=4	User supplied mixture of U-Pu-Zr fuel (input required at loc. numb. 5001+ & 5401+)
2			(Not currently used)
3	ICTYPE	=1	20% CW type 316 cladding
		=2	D9 cladding
		=3	HT9 cladding

continues on next page

Table 11.7.1 – continued from previous page

Location Number	Name	Value	Description
4			(Not currently used)
5	IDTOPT	=0	Compute at equally spaced time values (use NDT and DTIME)
		=1	Increments in time are user-supplied (use NDT and TTABLE)
6	IHTOPT	=0	Perform heat transfer calculation including coolant and wall
		=1	Perform heat transfer calculation with input values of clad outer surface temperature. (Use NCLADT and COMMON /TEMPIN/)
7-14		(8)	(Not currently used)
15	ICRACK	=0	No fuel cracking
		=1	Radial fuel cracks included
16	IFPLAS	=0	Allow creep-plastic strains in fuel
		=1	Suppress creep-plastic strains in fuel
17	ICPLAS	=0	Allow creep-plastic strains in clad
		=1	Suppress creep-plastic strain in clad
18	IFSWEL	=0	Allow swelling-hotpressing strains in fuel
		=1	Suppress swelling-hotpressing strains in fuel
19	ICSWEL	=0	Allow swelling strains in clad
		=1	Suppress swelling strains in clad
20	IDZ	=0	Axial segments are of equal height DZ=Z/NDZ
		=1	User-supplied axial segment heights
21	ILRGST	=0	Large strain analysis
		=1	Small perturbation analysis
22	IFCSLP	=0	Fuel-clad locked when gap is closed
		=1	Independent fuel-clad axial displacement
23			(Not currently used)
24	IOUTSW	=0	No detailed printing of results - summary of results only
		=1	Normal detailed printout under IFREQA, NFREQA, and IFREQB control
25	IFREQA		Initial print frequency, number of time steps between normal detailed printout
26	NFREQA		Total number of time steps under IFREQA control
27	IFREQB		Final print frequency
28	IGRAPH	=0	Do not write graphics file
		=1	Write datafile FT12F001 for processing by a graphics program
29-30		(2)	(Not currently used)
31	NDT		No. of time steps or no. of entries in TTABLE. If IDTOPT=0, there is no limitation on NDT. If IDTOPT=1, NDT is limited to 1980.
32	NDZ	(20)	No. of axial segments
33	NDRF	(20)	No. of radial elements in cladding (minimum value is 6)
34	NDRC	(10)	No. of radial elements in cladding (minimum value is 3)

continues on next page

Table 11.7.1 – continued from previous page

Location Number	Name	Value	Description
35-40		(6)	(Not currently used)
41	NQ	(25)	No. of values in power vs. time table
42	NCOOLT	(25)	No. of values in coolant inlet temperature versus time table
43	NCOOLF	(25)	No. of values in coolant inlet flow versus time table
44	NCLADT	(50)	No. of values in clad outer surface temperature versus time table
45-100		(56)	(Not currently used)
101-500	IETYPI(I,J)	(20,20)	Fuel element type. Required only when ICRACK=1, default element type is 1 (Loc. Numb. = 100+I+20*(J-1)).

*** Location Numbers for Decimal Data ***

(Symbols in parentheses after array variables are maximum storage allocation and name of integer variable specifying dimension. Numbers within quotation marks are recommended values)

Table 11.7.2: COMMON /CNTLIN/

Location Number	Name	Value	Description
1	TZERO		Initial time (sec)
2	DTIME	=1	Computation time step (if IDTOPT=0) (sec)
3-10		(8)	(Not currently used)
11-2000	TTABLE	(1990 NDT)	Computation time steps (if IDTOPT=1) (sec) maximum of 1990 values

Table 11.7.3: COMMON /GEOMIN/

Location Number	Name	Value	Description
2001	ZFUEL		Length of fuel column (cm)
2002-2010		(9)	(Not currently used)
2011-2030	DZ	(20 NDZ)	Height of axial segments (cm) (required only when IDZ=1)
2031-2050	FUELIR	(20 NDZ)	Inner radius of fuel (cm)
2051-2070	FUELOR	(20 NDZ)	Outer radius of fuel (cm)
2071-2090	CLADIR	(20 NDZ)	Inner radius of clad (cm)
2091-2110	CLADOR	(20 NDZ)	Outer radius of clad (cm)
(Note: values #2111-2112 not required when IHTOPT=1)			
2111	WALLIR		Inner radius of outer wall (cm)
2112	WALLOR		Outer radius of outer wall (cm)

continues on next page

Table 11.7.3 – continued from previous page

Location Number	Name	Value	Description
2113	ZPLENM		Total length of plenum (cm)
2114	ZPLNA		Length of sodium in plenum (cm)
2115	PLGASR		Gas constant for plenum gasses (Bar-cc/gm-K)
2116	PLTREF		Temperature at which PINT specified (K). Used to correct PINT to value consistent with initial plenum temperature.
2117-2200		(84)	(Not currently used)
2201-2600	FECTI(I,J)	(20,20 NDZ)	Initial radial crack strain in fuel (required only when ICRACK=1) (Loc. Numb.=2200+I+20*(J-1))
2601-3000		(400)	(Not currently used)

Table 11.7.4: COMMON /DRIVIN/

Location Number	Name	Value	Description
3001	PEXT		External pressure. Assumed constant during transient. (Bar)
3002	PINT		Initial value of internal pin press. (Bar)
3003-3004		(2)	(Not currently used)
(Note: values 3005-3006 not needed when IHTOPT=1)			
3005	TSINK		Temperature of outer heat sink (K)
3006	HSINK		Heat transfer coefficient-wall to outer heat sink (Watts/cm^2-K)
3007	QCONST		Energy generation constant. The energy generation rate for fuel radial segment I, axial segment J at time K is calculated as QCONST*QR(I)*QAX(J)*QT(K). The method of dimensioning and normalizing the four factors is arbitrary so long as the product dimension is in Watts/gm.
3008	BURNUP		Peak fuel burnup (atom %)
3009-3010		(2)	(Not currently used)
3011-3410	QR(I,J)	(20,20 NDZ)	Radial profile of energy generation rate. Values are required for NDRF radial elements in each axial segment. (See: QCONST for units) (Loc. Numb. = 3010+I+20*(J-1))
3411-3430	QAX	(20 NDZ)	Axial profile of energy generation rate (see: QCONST)
3431-3455	TQT	(25 NQ)	Time values in power vs. time table (sec)
3456-3480	QT	(25 NQ)	Power values in power vs. time table (see: QCONST)

continues on next page

Table 11.7.4 – continued from previous page

Location Number	Name	Value	Description
(Note: values 3481-3580 not needed when IHTOPT=1)			
3481-3505	TTIN	(25 NCOOLT)	Time values in coolant inlet temperature versus time table (sec)
3506-3530	TIN	(25 NCOOLT)	Coolant inlet temperatures (K)
3531-3555	TCFIN	(25 NCOOLF)	Time values in coolant inlet flow versus time table (sec)
3556-3580	CFIN	(25 NCOOLF)	Coolant inlet flow (gm/cm^2-sec)
3581-4000		(420)	(Not currently used)

Table 11.7.5: COMMON /MISCIN/

Location Number	Name	Value	Description
4001	GBFRAC		Fraction of FISGAS on grain boundaries (default value is 0.1)
4002-4039		(38)	(Not currently used)
4040	GASCON		Gas constant for central cavity gases (Bars-cc/gm-K)
4041-4060	CLDFLU	(20 NDZ)	Axial profile of clad fluence used in subroutine CFAIL (10^{22} n/cm^2)
4061-4080	CLDHRD	(20 NDZ)	Pre-transient hardness parameter used in clad flow stress calculation. (Default value is 0.223, the value appropriate to 20% CW unirradiated stainless steel.)
4081-4480	PORES(I,J)	(20,20 NDZ)	Dist. of total fuel porosity same radial grid as QR (do not include crack volume input) (Loc. Numb. = 4080+I+20*(J-1))
4481-4880	FISGAS(I,J)	(20,20 NDZ)	Dist. of fission gas (gm/cc) in fuel closed porosity and in solution on same radial grid as QR (do not include fission gas in open porosity). (Loc. Numb. = 4480+I+20*(J-1))
4881-5000		(120)	(Not currently used)
5001-5400	FRACPU(I,J)	(20,20 NDZ)	Dist. of plutonium (wt. %) in fuel (required when IFTYPE=4) (Loc. Numb. = 5000+I+20*(J-1))
5401-5800	FRACZR(I,J)	(20,20 NDZ)	Dist. of zirconium (wt. %) in fuel (required when IFTYPE=4) (Loc. Numb. = 5400+I+20*(J-1))
5801-6000		(200)	(Not currently used).

Table 11.7.6: COMMON /TEMPIN/

Location Number	Name	Value	Description
(Note: values 6001-7050 are only required when IHTOPT=1)			
6001-6050	TVALUE	(50 NCLADT)	Time values for clad surface temperature table (include TZERO) (sec)
6051-7050	TCSURF(J,K)	(20 NDZ,50 NCLADT)	Clad outer surface temperature for axial segment J at time K (K) (Loc. Numb. = 6050+J+20*(K-1))

*** Location Numbers for Debug Data ***

(All debug options default to zero. Therefore, input is required only if a debug option is to be used. Debug input is identified with location numbers larger than 9000 and it follows same conventions as regular input. Integer debug data is read in with regular integer data and decimal debug data is read in with regular decimal data.)

Table 11.7.7: COMMON /DEBUGI/ Integer Debug Data

Location Number	Name	Value	Description
9001	IDBOUT	=0	No debug output
		=1	Add debug output to regular IOUTSW=2 output
9002	IDBSTP	=0	Program stops when molten cavity freezes
		=1	Ignore this program stop
9003	IDBFPL	=0	Use recommended fuel flow stress Eq. (11.3-16)
		=1	Use simple power law fuel creep: $\dot{\varepsilon} = C_0 * \sigma_{\mathrm{e}}^{C_1}$ (Decimal values 9001 and 9002 are required)
9004	IDBFDV	=0	Use recommended fuel swelling - hotpressing (fuel swelling option for metal fuel is the simple grain boundary swelling model (ANL-IFR-27) & (ANL/RAS 83-33)
		=1	Use equilibrium swelling model (ANL-IFR-6 and -23)
		=2	Use simple power law fuel swelling: $\dot{\varepsilon} = C_0 * \sigma_{\mathrm{m}}^{C_1}$ (Decimal values 9003 & 9004 are required)
9005	IDBCPL	=0	Use recommended clad flow stress
		=1	Use ideal plastic flow for clad: $\sigma_{\mathrm{y}} = C_0 + C_1 \overline{\varepsilon}^p$ (Decimal values 9005 & 9006 are required)
		=2	Use high-temperature power-law creep
		=3	Use simple power law clad creep: $\dot{\varepsilon} = C_0 * \sigma_{\mathrm{e}}^{C_1}$ (Decimal values 9005 & 9006 are required)
9006	IGPRES	=0	Open fuel-clad gap pressure and plenum pressure remain at input values
		=1	User supplied gap pressure. This option allows FPIN2 to calculate pressure transients in gas pressurized cladding tubes. Plenum pressure is set equal to p_{gap}.

continues on next page

Table 11.7.7 – continued from previous page

Location Number	Name	Value	Description
9007	IGAPCL	=0	Use fuel-clad opening/closure model
		=1	Fuel-clad gap always closed
9008	ICPROP	=0	Use material property correlations
		=1	Use temperature independent material properties
9009	ISKIPM	=0	Perform complete thermal/mechanical calculations
		=1	Bypass mechanical calculation, heat trans. only
9010	IGCLOS	=0	Use gap closure routine at 100% fuel melting
		=1	Do not close gap (if open) at 100% fuel melting
9011-9050		(40)	(Not currently used)
9051	NGPRES	(25)	No. of values in p_{gap} vs. time table
9052-9100		(49)	(Not currently used)
9101-9120	IDBOTA	(20 NDZ)	Axial debug print vector (0=no prt, 1=prt)
9121-9140	IDBOTF	(20)	Fuel radial debug print vector
9141-9150	IDBOTC	(10)	Clad radial debug print vector

Table 11.7.8: COMMON /DEBUGD/ Decimal Debug Data

Location Number	Name	Value	Description
9001	FPLC0		Fuel power law creep constant C_0
9002	FPLC1		Fuel power law creep constant C_1
9003	FDVC0		Fuel power law swelling constant C_0
9004	FDVC1		Fuel power law swelling constant C_1
9005	CIPLC0		Clad idealized flow stress constant C_0
9006	CIPLC1		Clad idealized flow stress constant C_1
9007	HTERR		Relative convergence criterion for heat transfer calculation (ND) “0.0005”
9008	EPSCAV		Rel. convergence criterion for cavity pressure (ND) “0.001”
9009	EPSFE		Rel. convergence criterion for finite element analysis (ND) “0.0005”
9010	EPTEST		Rel. convergence criterion for plastic-creep strains (ND) “0.0005”
9011	EVTEST		Rel. convergence criterion for swelling strains (ND) “0.0005”
9012-9020		(9)	(Not currently used)
9021-9045	TGPRES	(25 NG PRES)	Time values in p_{gap} table
9046-9070	GPRES	(25 NGPRES)	Fuel-cladding gap pressure in p_{gap} table

Part IV: Failure Modeling

CHAPTER

TWELVE

COOLANT VOIDING

12.1 Introduction

The sodium voiding model in SAS4A/SASSYS-1 is a multiple-bubble slug ejection boiling model similar to that used in SAS2A *[12-1]*, SAS3A *[12-2]*, and SAS3D *[12-3]*, except that in SAS4A/SASSYS-1 changes have been made in the finite differencing to handle flow area changes and non-uniform axial nodes more consistently and accurately. Another difference is that in addition to voiding due to boiling, SAS4A/SASSYS-1 will also calculate voiding due to plenum gas release from failed fuel pins. The main purposes of this model are to predict the rate and extent of voiding for the voiding reactivity calculations and to predict the heat removal from the cladding surface, after the onset of voiding, for the fuel and cladding temperature calculations.

Figure 12.1.1 illustrates the voiding model. Each channel in SAS4A/SASSYS-1 represents a fuel pin and its associated coolant and structure. Voiding is assumed to result in the formation of bubbles that fill the whole cross section of the coolant channel except for a film of liquid sodium that is left on the cladding and on the structure. Up to nine bubbles, separated by liquid slugs, are allowed in the channel at any time.

Currently, the film is treated as a static film of a thickness that changes due to vaporization or condensation. A dynamic film treatment in which film motion is determined by the combined effects of gravity and the shear force of streaming vapor will be added to a future version of the code.

The extent of voiding is determined mainly by liquid slug motion. The bubble pressures at the bubble-slug interfaces drive the liquid slugs. Therefore, the voiding calculation couples vapor or gas pressure calculations for the bubbles with momentum equations for the liquid slugs. If a bubble is small, it is assumed that the vapor or gas pressure within the bubble is constant throughout the bubble, and the bubble pressure is computed using the uniform vapor pressure model described in Section 12.5. For larger bubbles, the vapor pressure is calculated from the pressure gradient model explained in Section 12.6. Section 12.7 describes the gas voiding model.

The voiding model portrayed in Fig. 12.1-2 has been developed and extensively tested throughout the evolution of the SAS family of codes. A considerable body of experimental information dictated the choice of a multi-bubble slug ejection model. The justification for using a one-dimensional, slug flow model first came from stimulant material experiments by Grolmes and Fauske *[12-4]*. These tests indicated that voiding took place under slug flow conditions, with a small liquid film left on the cladding and structure. The multiple bubble, slug flow model has been compared with results from a variety of out-of-pile and in-pile experiments, including tests at Argonne with the OPERA, TREAT, and SLSF facilities, experiments conduced at KFK *[12-5]*, and the ORNL Sodium Boiling Tests *[12-6]*. These comparisons have consistently shown that the SAS4A/SASSYS-1 voiding model agrees well with experimental data that are prototypic of reactor conditions.

When the voiding model is translated into a computer code, the result is a rather lengthy series of subroutines, which, at first glance, appears to be intimidatingly complex. However, this appearance is somewhat misleading. The voiding model coding seems less intricate if attention is focused initially on just those subroutines that perform the major calculations needed by the model. There are three such major calculations; in order, they are

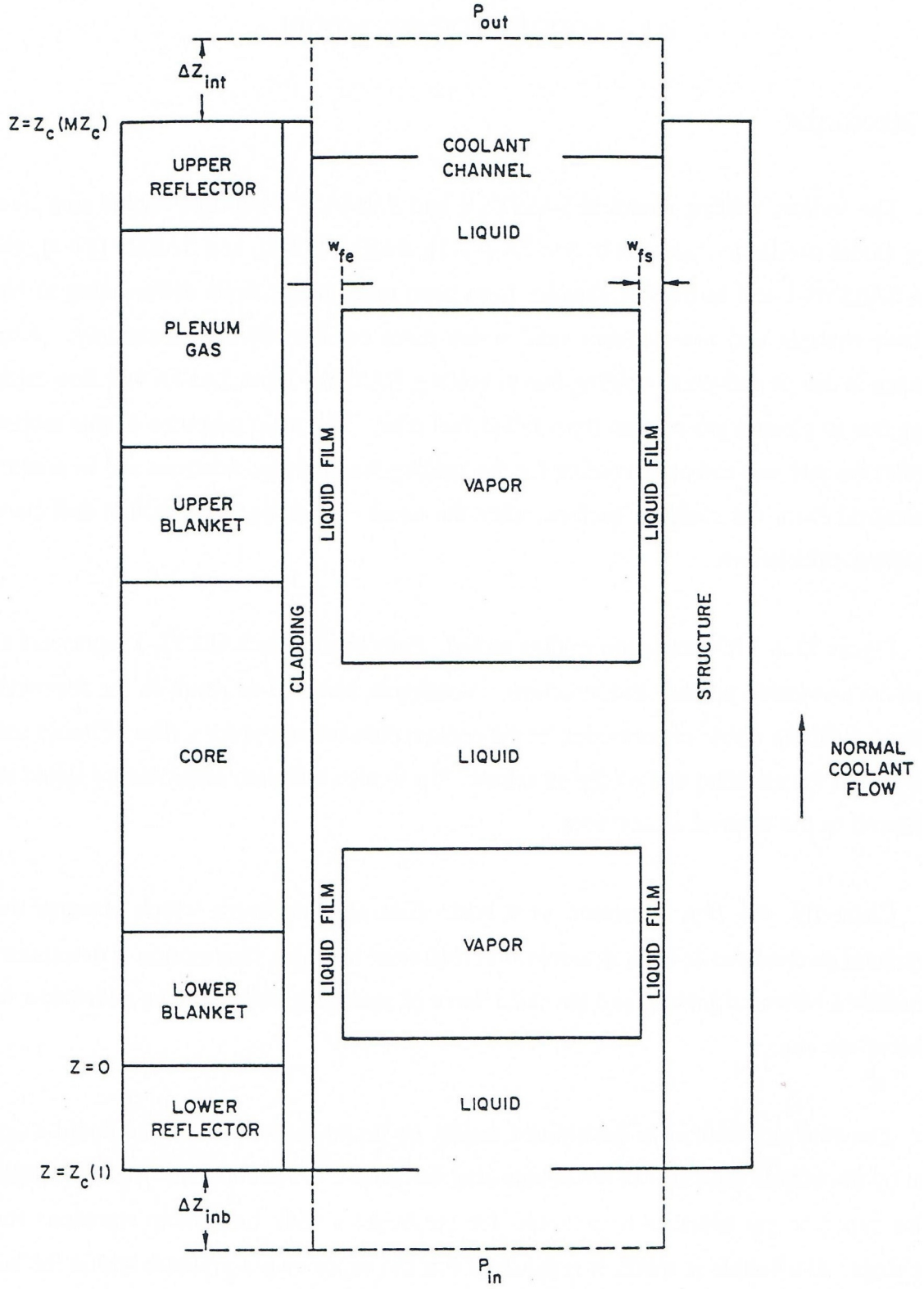


Figure 12.1.1: Voiding Model Geometry

1. a preliminary liquid slug flow calculation that neglects the effect of changes in the vapor bubble pressures over the time step (performed in subroutine TSC2),
2. the calculation of the advanced time pressures and mass flow rates in the vapor bubbles (performed in subroutine T4A3D), and
3. a liquid slug flow calculation that accounts for the updated bubble conditions (performed in subroutines TSC5, TSC6, and TSC7).

The remaining voiding model subroutines perform computations that either directly or indirectly support the three major calculations just listed (i.e., initialization of the calculation in TSCSET, determination of new time-step size in TSC9, etc.). A detailed description of all subroutines and the overall flow of the coding will be given in Section 12.10.

The relationship of the voiding model to other major modules in SASSYS-1 is diagrammed in Figure 12.2.1. This drawing shows that the voiding model picks up its initial information from the transient heat-transfer modules, supplies information to the point kinetics module (which, in turn, feeds data to the transient heat-transfer module), and provides clad surface heat fluxes to the transient heat transfer module. Thus, the voiding model has a direct impact on most of the major modules in SASSYS-1. Further details of the interaction of the voiding model with other modes in the code will be presented in Section 12.9.

The remainder of the chapter is divided into five major categories: calculation of mass flow rates in the liquid slugs, calculation of vapor pressures in small bubbles through the uniform vapor pressure model, calculation of vapor pressures and mass flow rates in larger bubbles by the vapor pressure gradient model, the gas voiding model, and discussion of the coding itself. Section 12.2 and Section 12.3 cover the first category of liquid slug calculations, Section 12.4 introduces the topic of vapor bubble calculations, Section 12.5 discusses the uniform vapor pressure model, Section 12.6 explains the pressure gradient model, Section 12.7 describes the gas voiding model and Section 12.8 through *12.13* discuss various aspects of the voiding model coding, including physical properties correlations.

12.2 Liquid Slug Flow Rates

The mass flow rate in each of the individual slugs pictured in Figure 12.1.1 is computed from the momentum equation similarly to the calculation of the prevoiding mass flow rate described in Chapter 3. Because of this similarity, the discussion here will be brief and will emphasize the additions that must be made to the model of Chapter 3 to accommodate the fact that there are several separate liquid slugs in the channel rather than one slug which fills the channel as in the prevoiding case. In particular, the model must now be able to handle the fact that the liquid slugs are continually changing length and must account properly for the effect of the bubble vapor pressures in driving the liquid slug flow rates.

The foundation of the model is again the liquid momentum equation, Eq. (3.9-1):

$$\frac{1}{A_c}\frac{\partial W}{\partial t}+\frac{\partial p}{\partial z}+\frac{1}{A_c}\frac{\partial (Wv)}{\partial z}=-\left(\frac{\partial p}{\partial z}\right)_{fr}-\left(\frac{\partial p}{\partial z}\right)_{K}-\rho_c g \tag{12.2-1}$$

with the terms as defined in Chapter 3. Now, however, the momentum equation is applied to each slug individually and is integrated over the length of each slug rather than over the length of the channel, giving an integral equation of the form of Eq. (3.9-8):

$$I_1\frac{\partial W}{\partial t}+p_t-p_b+W^2 I_2+A_{fr}W\left|W\right|^{1+b_{fr}}I_3+W\left|W\right|I_4+gI_5=0 \tag{12.2-2}$$

where

$$I_1=\int\frac{dz}{A_c}=\sum_{JC=JST}^{JEND}X_{I1}(JC) \tag{12.2-3}$$

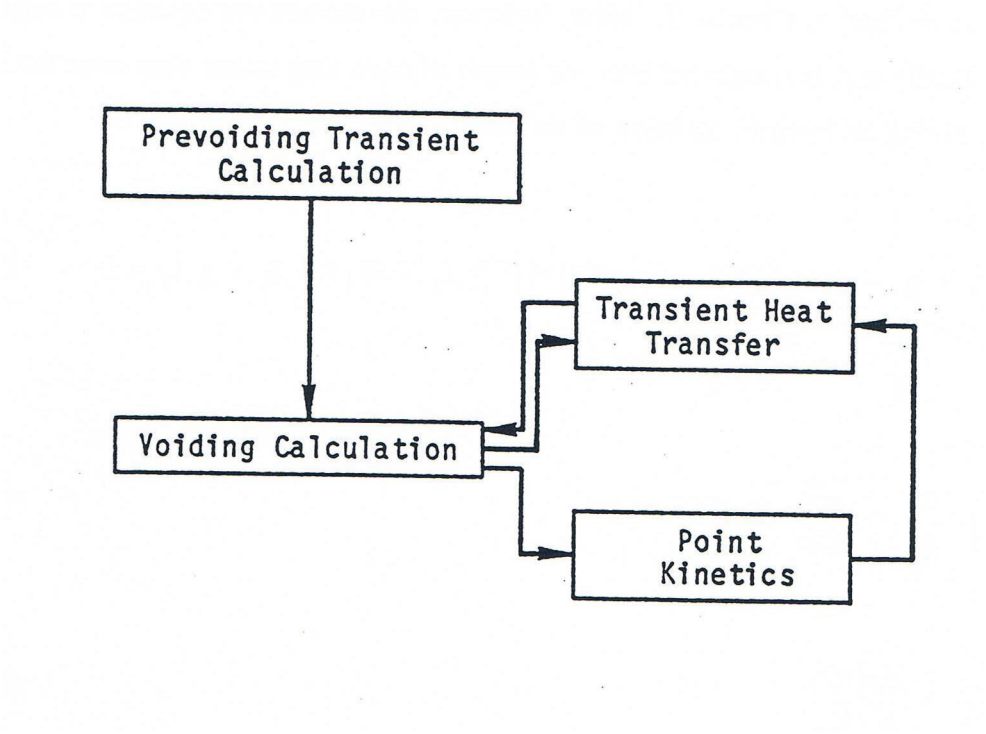


Figure 12.2.1: Schematic Drawing of the Relationship of the Voiding Model to Other SASSYS-1 Models

$$X_{I1}\,(\text{JC}) = \frac{\Delta z(JC)}{A_c\,(\text{JC})} \tag{12.2-4}$$

$$I_2 = \sum_{JC=JST}^{JEND} X_{I2}(JC) \tag{12.2-5}$$

$$X_{I2}\,(JC) = \frac{1}{A_c\,(JC)^2}\left[\frac{1}{\rho_c(JC+1)} - \frac{1}{\rho_c(JC)}\right] \tag{12.2-6}$$

$$I_3 = \int \frac{1}{2\rho_c A_c^2 D_h}\left[\frac{D_h}{\mu A_c}\right]^{b_{fr}} dz = \sum_{JC=JST}^{JEND} X_{I3}(JC) \tag{12.2-7}$$

$$X_{I3}\,(JC) = \frac{\Delta z(JC)}{\left[\rho_c\,(JC) + \rho_c\,(JC+1)\right] A_c\,(JC)^2\, D_h(JC)}\left[\frac{D_h\,(\text{JC})}{\overline{\mu}\,(JC)\,A_c\,(JC)}\right]^{b_{fr}} \tag{12.2-8}$$

$$I_4 = \sum_{JC=JST}^{\text{JEND}} K_{or}(JC) \tag{12.2-9}$$

$$I_5 = \int \rho_c dx = \sum_{JC=JST}^{\text{JEND}} X_{I5}(JC) \tag{12.2-10}$$

$$X_{I5}\,(\text{JC}) = 0.5\left[\rho_c\,(\text{JC}) + \rho_c\,(JC+1)\right]\Delta z(JC) \tag{12.2-11}$$

where

P_b = the pressure at the bottom of the slug,

and

P_t = the pressure at the top of the slug.

The variable JST is the number of the mesh segment in which the bottom of the liquid slug is located, while JEND is the number of the segment in which the top is contained. The liquid slug configuration in the SASSYS-1 axial mesh is shown in Figure 12.2.1. Note that mesh segments JST and JEND are part liquid and part vapor. Since the integration is over only the liquid portions of these segments, the axial length terms $\Delta z\,(JC)$ in the expressions for X_{I1}, X_{I3} and X_{I5} must be altered for segments JST and AJEND from being the mesh segment height to being the length of the portion of the segment which is filled with liquid. This is represented by the term $\Delta z'$

$$\text{If } K > 1,\ \Delta z^{'}\,(JST) = z_c\,(JST+1) - z_i(L=1,t,K-1) \tag{12.2-12}$$

$$\text{If } K = 1,\ \Delta z^{'}\,(JST) = \Delta z(JST) \tag{12.2-13}$$

$$\text{If } K \le K_{vn},\ \Delta z^{'}\,(JEND) = z_i\,(L=1,t,k) - z_c(JEND) \tag{12.2-14}$$

$$\text{If } K > K_{vn},\ \Delta\,(JEND) = \Delta z(JEND) \tag{12.2-15}$$

The notation $z_i(L,t,K)$ for the interface position is derived from the coding and is to be interpreted as follows. The quantity L takes the value 1 or 2, with $L = 1$ denoting the lower interface of bubble K and $L = 1$ the upper interface. The variable t is simply time, and K is just the bubble number, with the numbering going from 1 for the lowest bubble in the channel to K_{vn} for the highest bubble.

Two other modifications must be made to accommodate the partial segments at the ends of the slug. First, the density $p_c(JST)$ in the expressions for $X_{I2}(JST)$ and $X_{I3}(JST)$ must be replaced by the liquid density at the bubble interface; similarly, $\rho_c(JEND+1)$ must be replaced by the interface value in $X_{I2}(JEND)$ and $X_{I3}(JEND)$. Second, the terms for JST and JEND in the computation of I_4 must be multiplied by the fraction $\Delta z'/\Delta z$.

If JST = 1, the effective inertial term $[\frac{\Delta z_i}{A}]_b$ must be added to I_1, and if JEND = MZC, the term $[\frac{\Delta z_i}{A}]_t$ must be included in I_1, just as in Chapter 3, Eq. (3.9-9).

In the special case of a small liquid slug entirely contained within one mesh segment, JST = JEND. Then,

$$\Delta z^{'}\,(\text{JST}) = z_i\,(L=1,t,K) - z_i(L=2,t,K-1) \tag{12.2-16}$$

and there is only one term in each of the summations for I_1 through I_5.

Eq. (12.2-2) must now be finite differenced and solved for the liquid mass flow rate in the slug. The advanced time mass flow rate W is forward differenced in time, and functions of W in Eq. (12.2-2) are expressed using first-order Taylor series; see Eq. (3.9-26) through Eq. (3.9-33) for the detailed expressions. However, the differenced form of Eq. (12.2-2) is more complex than the corresponding equation in the prevoiding model, Eq. (3.9.4), because now the coefficients I_1 through I_5 are functions of time due to motion of the liquid-vapor interfaces, and, therefore, the time level at which these coefficients are evaluated must be specified. To allow flexibility in the level of implicitness of the calculation, the variable explicit/implicit scheme used in Eq. (3.9-26) is applied to the coefficients in Eq. (12.2-2). The differenced equation is then

$$\begin{aligned}
&\left(\theta_1 I_1|_t + \theta_2 I_1|_{t+\Delta t}\right)\frac{\Delta W}{\Delta t} + \theta_1\left(p_{t_1} - p_{b_1}\right) + \theta_2\left(p_{t_2} - p_{b_2}\right)\\
&+\left(\theta_1 I_2|_t + \theta_2 I_2|_{t+\Delta t}\right)\left(W_1^2 + 2\theta_2 W_1 \Delta W\right)\\
&+A_{fr}\left(\theta_1 I_3|_t + \theta_2 I_3|_{t+\Delta t}\right)\left[W_1|W_1|^{1+b_{fr}}\right.\\
&\left.+\theta_2\left(2+b_{fr}\right)|W_1|^{1+b_{fr}}\Delta W\right]\\
&+\left(\theta_1 I_4|_t + \theta_2 I_4|_{t+\Delta t}\right)\left(W_1|W_1| + 2\theta_2|W_1|\Delta W\right)\\
&+\left(\theta_1 I_4|_t + \theta_2 I_4|_{t+\Delta t}\right) g = 0
\end{aligned} \tag{12.2-17}$$

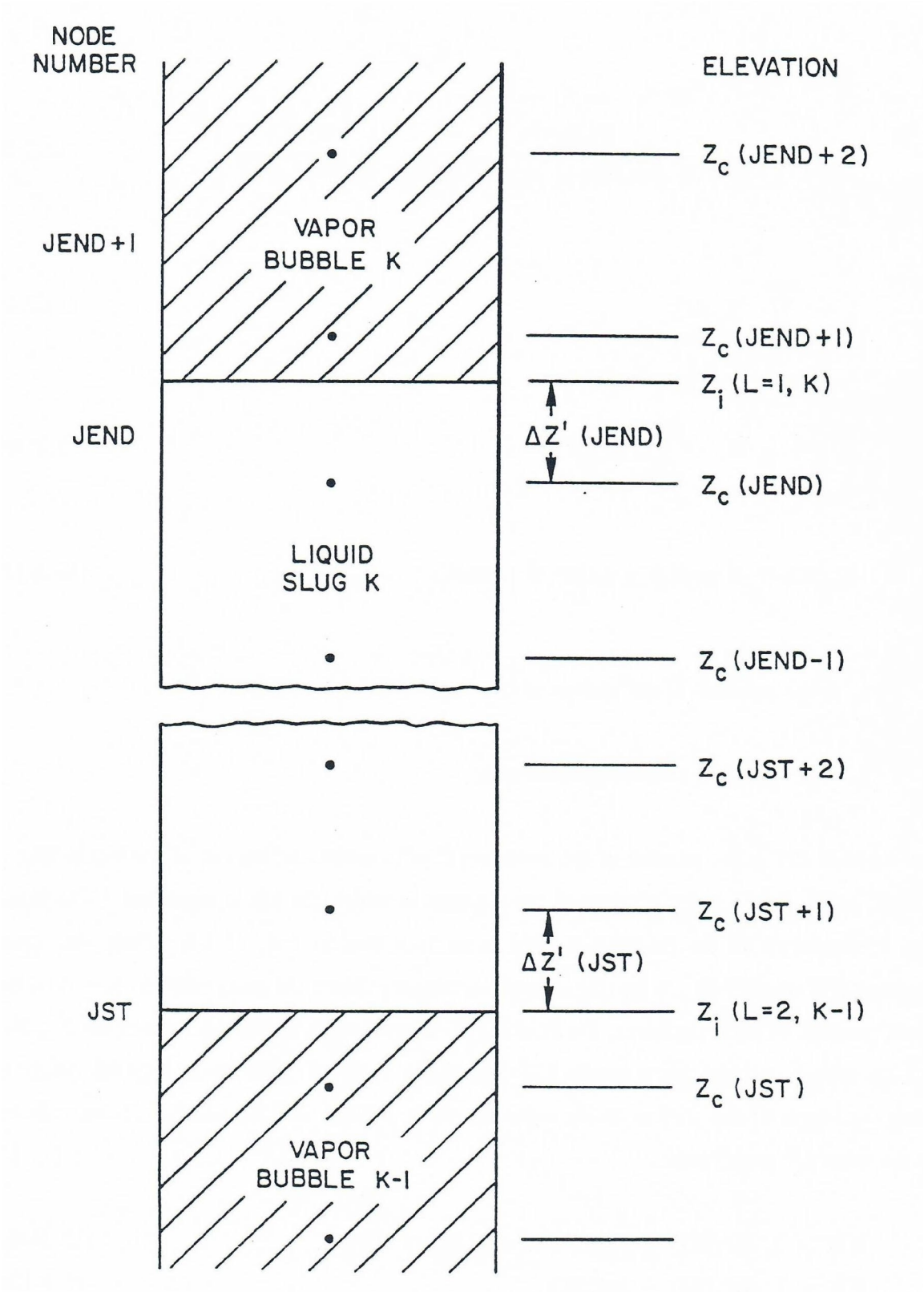


Figure 12.2.2: Placement of a Liquid Slug in the SASSYS-1 Axial Coolant Mesh

The change in the coefficient I_2 over a time step is a function only of the change in interface liquid density, which is a very small effect and can be neglected. Therefore, I_2 is considered constant over the time step. In addition, the orifice coefficient I_4 is also assumed constant over the time step. The advanced time values of the other three coefficients are expressed using forward differencing in time. To illustrate, take the gravity coefficient I_5; at time t,

$$I_5\left(t\right) = \int_{Z_{\text{JST}}(t)}^{Z_{\text{JEND}}(t)} \rho_l\left(z\right) \text{dz} \tag{12.2-18}$$

and at time $t + \Delta t$,

$$I_5\left(t+\Delta t\right) = \int_{Z_{\text{JST}}(t+\Delta t)}^{Z_{\text{JEND}}(t+\Delta t)} \rho_l\left(z\right) \text{dz} \tag{12.2-19}$$

These two integrals are identical except in segments JST and JEND, where the bubble interface positions are changing with time. Taking the difference gives

$$I_5\left(t+\Delta t\right) - I_5\left(t\right) = \int_{Z_{\text{JEND}}(t)}^{Z_{\text{JEND}}(t+\Delta t)} \rho_l\left(z\right) \text{dt} - \int_{Z_{\text{JST}}(t)}^{Z_{\text{JST}}(t+\Delta t)} \rho_l\left(z\right) \text{dz} \tag{12.2-20}$$

The interface position z_i can be written as a linear function of the interface velocity v_i, so that

$$dz = v_i dt \tag{12.2-21}$$

Therefore, $I_5\left(t+\Delta t\right)$ is

$$I_5\left(t+\Delta t\right) = I_5\left(t\right) + \rho_{li}\left(\text{JEND}\right) \int_t^{t+\Delta t} v_i\left(L=1, t^{'}, K\right) dt^{'} - \rho_{li}(JST) \int_t^{t+\Delta t} v_i\left(L=2, t^{'}, K-1\right) dt' \tag{12.2-22}$$

If the velocity is assumed to vary linearly over the time step, the time integrals in Eq. (12.2-22) become

$$\begin{aligned} \int_t^{t+\Delta t} v_i\left(t'\right) dt' &= \overline{v}\Delta t \\ &= \left[v_i\left(t\right) + \frac{\Delta v}{2 v_i\left(t\right)}\right] \Delta t \\ &= v_i\left(t\right) \left[1 + \frac{\Delta v}{2 v_i\left(t\right)}\right] \Delta t \\ &= v_i\left(t\right) \left[1 + \frac{\Delta W}{2 W_1}\right] \Delta t \end{aligned} \tag{12.2-23}$$

where $\Delta v = v_i\left(t+\Delta t\right) - v_i\left(t\right)$. Therefore, $I_5\left(t+\Delta t\right)$ is

$$\begin{aligned} I_5\left(t+\Delta t\right) = I_5\left(t\right) &+ \rho_{li}\left(\text{JEND}\right) v_i\left(L=1, t, K\right) \left[1 + \frac{\Delta W}{2 W_1}\right] \Delta t \\ &- \rho_{li}\left(\text{JST}\right) v_i\left(L=2, t, K-1\right) \left[1 + \frac{\Delta W}{2 W_1}\right] \Delta t \end{aligned} \tag{12.2-24}$$

This expression is valid for a liquid slug that is bounded at both ends by vapor bubbles. However, slugs that extend out the top of the subassembly do not have a moving upper interface, and so the integral over the upper interface in Eq. (12.2-20) is zero. Similarly, the integral over the lower interface is zero for slugs extending out the bottom of the subassembly. To preserve the ease of using one expression for $I_5\left(t+\Delta t\right)$ for all types of liquid slugs, Eq. (12.2-24) is rewritten as

$$I_5\left(t+\Delta t\right) = I_5\left(t\right) + \Delta I_5 + \Delta W I_5' \tag{12.2-25}$$

where

$$\Delta I_5 = \Delta t v_i \left(L = 1, t, K\right) \rho_{li} \left(\text{JEND}\right) x_u - \Delta t v_i \left(L = 2, t, K-1\right) \rho_{li} \left(\text{JST}\right) x_L \tag{12.2-26}$$

$$I_5^{'} = \frac{\Delta I_5}{2W_1} \tag{12.2-27}$$

$$x_L = 1, \text{ for slug types 2 or 3} \tag{12.2-28}$$

$$x_L = 0, \text{ otherwise} \tag{12.2-29}$$

$$x_u = 1, \text{ for slug types 1 or 2} \tag{12.2-30}$$

$$x_u = 0, \text{ otherwise} \tag{12.2-31}$$

Slug type 1 refers to a slug that extends below the subassembly inlet, type 2 to a slug that is bounded top and bottom by bubbles, and type 3 to a slug that extends out the subassembly Table 12.2.1 defines all slug-type numbers used in the code, as well as listing the bubble-type numbers that are part of the coding. Note that the increment ΔI_5 represents the change I_5 if the change in the mass flow rate of the slug is ignored, while the I_5' term accounts for the change due to acceleration of the slug.

The expressions for $I_1\left(t+\Delta t\right)$ and $I_3\left(t+\Delta t\right)$ are derived in the same way as was Eq. (12.2-25). The resulting equations are

$$I_1\left(t+\Delta t\right) = I_1\left(t\right) + \Delta I_1 \tag{12.2-32}$$

where

$$\Delta I_1 = \frac{\Delta t v_i \left(L = 1, t, K\right) x_u}{A_c(JEND)} - \frac{\Delta t v_i \left(L = 2, t, K-1\right) x_L}{A_c(JST)} \tag{12.2-33}$$

$$I_3\left(t+\Delta t\right) = I_3\left(t\right) + \Delta I_3 + \Delta W I_3^{'} \tag{12.2-34}$$

with

$$\begin{aligned}\Delta I_3 = &\frac{\Delta t v_i \left(L = 1, t, K\right) x_u}{2A_c\left(\text{JEND}\right)^2 \rho_{lt}\left(K\right) D_h(JEND)} \left[\frac{D_h\left(\text{JEND}\right)}{A_c\left(\text{JEND}\right)\overline{\mu}\left(\text{JEND}\right)}\right]^{b_{fr}} \\ &- \frac{\Delta t v_i \left(L = 2, t, K-1\right) x_L}{2A_c\left(\text{JST}\right)^2 \rho_{lb}\left(K\right) D_h\left(\text{JST}\right)} \left[\frac{D_h\left(\text{JST}\right)}{A_c(JST)\overline{\mu}\left(\text{JST}\right)}\right]^{b_{fr}}\end{aligned} \tag{12.2-35}$$

and

$$I_3^{'} = \frac{\Delta I_3}{2W_1} \tag{12.2-36}$$

The effect of slug acceleration is omitted in $I_1\left(t+\Delta t\right)$.

Table 12.2.1: Liquid Slug Types and Bubble Types Used in Coolant Dynamics Model

	1. Bubble Types
	The FORTRAN variable IBUB1 and IBUB2 refer to bubble types.
IBUB	Bubble Type
1	Bubble within the subassembly; uniform vapor pressure bubble.
2	Bubble within the subassembly; pressure gradient model.
3	Bubble extends out the top of the subassembly; uniform vapor pressure bubble.
4	Bubble extends out the top of the subassembly; pressure gradient model.
5	Bubble extends out both the top and bottom of the subassembly.
6	Bubble extends out the bottom of the subassembly.
	2. Liquid Slug Types
	The FORTRAN variables ILIQ1 and ILIQ2 refer to liquid slug type.
ILIQ	Liquid Slug Type
1	Lower liquid slug, below the lowest bubble.
2	Intermediate liquid slug, with bubbles above and below.
3	Upper liquid slug, above the highest bubble.
4	Liquid slug fills whole subassembly; no voiding.
5	Lower liquid slug below the subassembly inlet, below a type-5 or -6 bubble.
6	Upper liquid slug above the subassembly outlet, above a type-3, -4, or -5 bubble.

Substituting the above expressions for the advanced time coefficients into Eq. (12.2-17) and neglecting second-order terms produces the differenced momentum equation

$$\begin{aligned}(I_1+\theta_2\Delta I_1)\frac{\Delta W}{\Delta t} = {} & \theta_1(p_{b1}-p_{t1})+\theta_2(p_{b2}-p_{t2})-I_2W_1^2 \\ & -2\theta_2\Delta W W_1 I_2 - A_{fr}(I_3+\theta_2\Delta I_3)W_1|W_1|^{1+b_{fr}} \\ & -(2+b_{fr})\theta_2 A_{fr} I_3|W_1|^{1+b_{fr}}\Delta W \\ & -\theta_2 A_{fr} W_1 I_3^{'}|W_1|^{1+b_{fr}}\Delta W - I_4(W_1|W_1|+2\theta_2 W_1\Delta W) \\ & -g(I_5+\theta_2\Delta I_5)-gI_5^{'}\theta_2\Delta W\end{aligned} \tag{12.2-37}$$

which gives

$$\Delta W = \frac{\Delta t[AA_0+(\Delta p_b-\Delta p_t)\theta_2]}{I_1+\theta_2\Delta I_1+\theta_2 BB_0\Delta t} \tag{12.2-38}$$

where

$$AA_0 = p_{b1}-p_{t1}-I_2W_1^2-A_{fr}(I_3+\theta_2\Delta I_3)W_1|W_1|^{1+b_{fr}}-I_4W_1|W_1|-g(I_5+\theta_2\Delta I_5) \tag{12.2-39}$$

and

$$BB_0 = 2W_1I_2+(2+b_{fr})A_{fr}I_3|W_1|^{1+b_{fr}}+2I_4|W_1|+A_{fr}|W_1|^{1+b_{fr}}W_1I_3^{'}+gI_5^{'} \tag{12.2-40}$$

Thus the change in flow rate for a liquid slug is related to the changes in vapor pressures in the bubbles above and below the liquid slug, or to the changes in inlet and outlet coolant pressures.

12.3 Interface Velocities

The average velocity in a liquid slug is

$$v_l = \frac{W}{\rho_l A_c} \tag{12.3-1}$$

but the presence of films on the cladding and structure causes the liquid-vapor interface to move at a somewhat different velocity. Figure 12.3.1 shows an interface moving at a velocity v_i for a time interval Δt. A film of liquid sodium is present in the vapor region on both the cladding and structure. The cladding film is of thickness w_{fe}, while that on the structure has thickness w_{fs}. The film on the cladding can move with velocity v_{fe}, and the structure film can have a velocity of v_{fs}. In later versions of SASSYS-1, film motion modeling will be available, and so v_{fe} and v_{fs} are included in the equations below for completeness; currently, however, film motion is neglected, and v_{fe} and v_{fs} are set to zero.

If the coolant channel volume between $z_i\,(t)$ and $z_i\,(t+\Delta t)$ is taken as a control volume, then the liquid volume V_l in the control volume at time t is

$$V_l\,(t) = P_e v_i \Delta t (w_{fe} + \gamma_2 w_{fs}) \tag{12.3-2}$$

where P_e is the outer perimeter of the cladding and γ_2 is the ratio of the surface area of the structure to the surface area of the cladding. At $t+\Delta t$ the liquid volume in the control volume is

$$V_l\,(t+\Delta t) = A_c v_i \Delta t \tag{12.3-3}$$

Accounting for the liquid added to and subtracted from the control volume during Δt gives

$$V_l\,(t+\Delta t) = V_l\,(t) + v_l A_c \Delta t - P_e\,(w_{fe} v_{fe} + \gamma_2 w_{fs} v_{fs})\,\Delta t \tag{12.3-4}$$

Substituting Eq. (12.3-2) and Eq. (12.3-3) into Eq. (12.3-4) gives

$$A_c v_i \Delta t = P_e v_i \Delta t\,(w_{fe} + \gamma_2 w_{fs}) + v_l A_c \Delta t - P_e \Delta t (w_{fe} v_{fe} + \gamma_2 w_{fs} v_{fs}) \tag{12.3-5}$$

or

$$v_i = \frac{v_l - P_e\,(w_{fe} v_{fe} + \gamma_2 w_{fs} v_{fs})\,/A_c}{1 - P_e (w_{fe} + \gamma_2 w_{fs})/A_c}, \tag{12.3-6}$$

which is the equation used for computing interface velocities. Note that this equation is applicable whether the interface is moving in the positive or the negative axial direction and holds whether the interface is moving towards the liquid side or the vapor side.

12.4 Bubble Formation and Collapse

At the end of each time step before voiding starts in a channel, the coolant temperature at each axial node is compared with the corresponding saturation temperature. The code estimates when boiling will start by linearly extrapolating coolant temperatures and saturation temperatures. A switch to the boiling module is made before the formation of the first bubble. The voiding model then takes over the calculation of coolant temperatures and saturation temperatures by applying the liquid slug model described in Section 12.2 to the entire channel. At the end of each time step, the code checks to see whether the boiling criterion has been met at any point along the axial mesh. This criterion requires that the coolant temperature exceed the saturation temperature by a minimum superheat value. This superheat value is selected by the user and is entered through input variable DTS. Experience with the code has shown that a superheat value of about 10 K is appropriate for the initial bubble

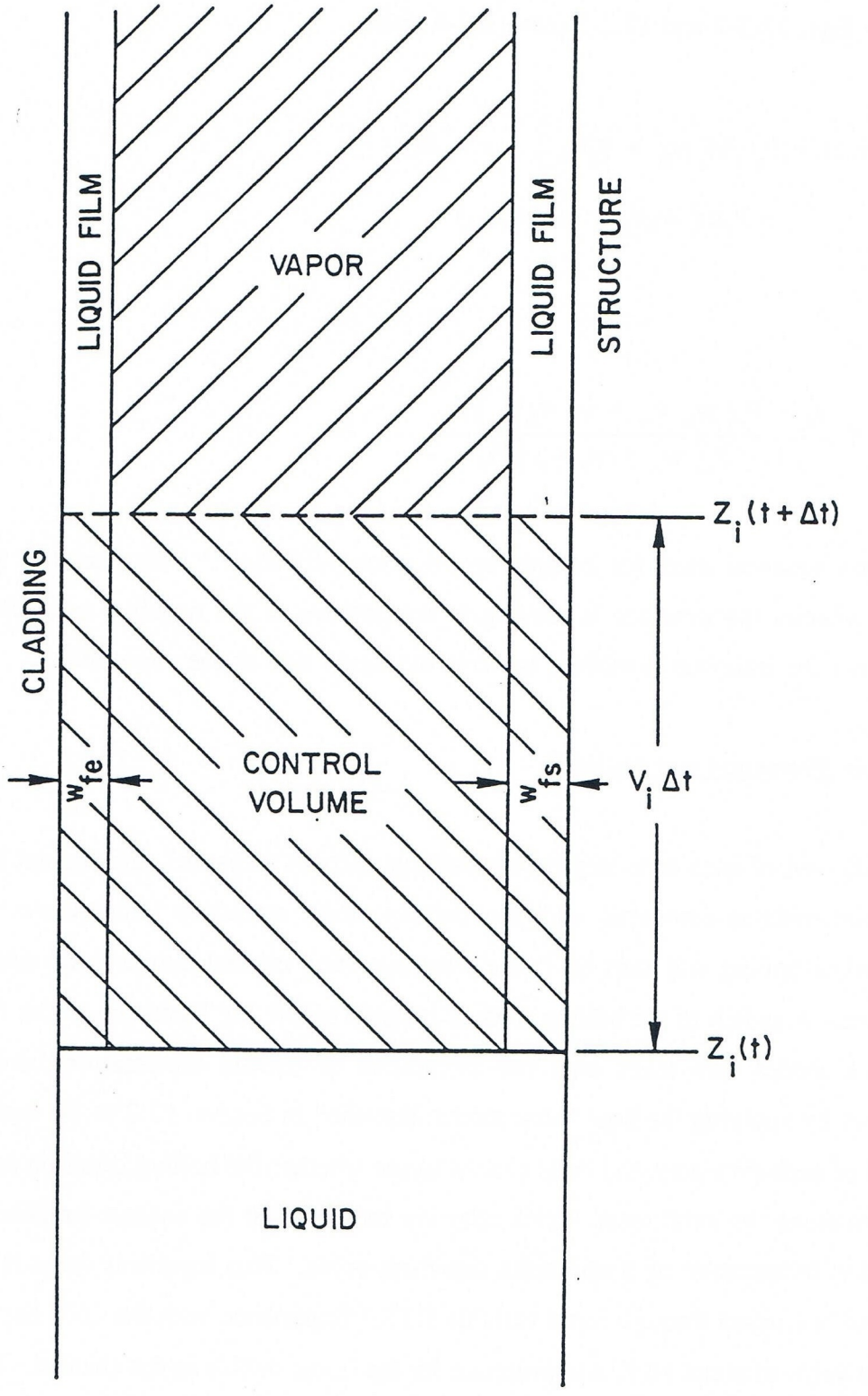


Figure 12.3.1: Geometry Associated with the Liquid-Vapor Interface Velocity Calculation

in the channel. This amount of superheat prevents the bubble from collapsing immediately after formation and appears to produce results that agree well with the experimental results discussed in the introduction to the chapter.

Once the code predicts that the coolant temperature has exceeded the saturation temperature plus the superheat, it begins an iterative process to determine within a small tolerance the time at which the superheat criterion is satisfied exactly. This is done by checking to see by how much the coolant temperature exceeds the sum of the saturation temperature plus the superheat. This temperature excess is used in a linear interpolation to reduce the time step to a value at which the superheat criterion will be met more closely. The code then goes back to the beginning of the time step and repeats the coolant and saturation temperature calculations and again checks the superheat criterion. If the criterion is met to within a built-in tolerance (0.001 K), the code fixes that time as the time of voiding initiation for the channel and forms a vapor bubble at that point in the channel. If the criterion is not met, the code repeats the calculation of the temperature excess beyond the superheat criterion, calculates a new reduced time step, and again goes back to the beginning of the time step and repeats the temperature calculations. This iterative procedure continues until the superheat criterion is met satisfactorily. Normally, the procedure converges with three or four iterations.

After voiding has started in a channel, the voiding model calculation tests at the end of each coolant time step for the formation of new bubbles within the liquid slugs. The formation of new bubbles after voiding has started is subject to the following limitations and modifications.

a. If the maximum number of bubbles (nine bubbles) already exist in the channel, then new bubbles will not be formed.

b. No new bubbles will be formed within a minimum distance, s_{Lmin} (minimum slug length) of a bubble-liquid interface, so nodes within s_{Lmin} of an interface are not checked for bubble formation. Interpolated coolant temperatures and saturation temperatures at a distance s_{Lmin} from each interface are checked, and a bubble is formed at this position if the superheat criterion is exceeded. The minimum distance s_{Lmin} is an input variable **SLMIN**; experience with the code indicates that a value of about 2 cm is reasonable for most situations.

c. No more than one new bubble will be formed in a channel in any time step.

d. After the first bubble is formed, time steps are not repeated so as to match exactly the superheat criterion for later bubbles; when the superheat criterion is exceeded, a new bubble is formed with whatever superheat happens to exist. The superheat used in the superheat criterion for bubbles after the first one should be somewhat less than the value used at the start of boiling, generally 3 K or 4 K. This is also a user-input quantity and is read into input variable **DTSI**. At the end of each time step, linear extrapolation of coolant and saturation temperatures is used to predict when the next bubble will form. This prediction is used to limit the size of future time steps, so that the saturation criterion is normally exceeded only slightly when a new bubble is formed.

Bubbles are assumed to fill the whole cross section of the coolant channel, except for a liquid film left on the cladding and structure. A bubble starts with zero length and with the initial temperature equal to the coolant liquid temperature at the time of bubble formation.

Pressures are continuous across the liquid-vapor interfaces; but sharp temperature gradients can occur in the liquid temperature a short distance from the interface can be significantly different from the vapor temperature at the interface. For each bubble interface, the code calculates one pressure, a vapor temperature, and a liquid temperature at a position far enough from the interface that it is unaffected by the vapor temperature. Also, the bubble interface position and velocity are calculated. In addition to bubble interface values, coolant temperatures, pressures, densities, and flow rate are calculated at the mesh segment interfaces. The liquid film thickness in a voided region is calculated as the average for the segment. Flow areas and hydraulic diameters are treated as constant within a segment although they can vary from segment to segment.

The vapor bubble pressure is assumed to be always equal to the saturation pressure corresponding to its temperature. Thus, the formation of a vapor bubble with a nonzero amount of superheat leads to an immediate jump

in pressure at the bubble location.

The initial bubble growth is due mainly to two effects: (1) the initial jump in pressure corresponding to the superheat, drives the liquid slugs apart, forming a larger bubble; and (2) heat flow through the liquid-vapor interface produces vapor to fill the bubble and sustain the pressure. When the bubble gets larger, vaporization of the liquid film on the cladding becomes the main source of vapor. The vapor pressure and temperature are assumed to be spatially uniform during the initial bubble growth. This model for the initial bubble growth is similar to the mode of Fauske and Cronenberg *[12-7, 12-8]* and also similar to that of Schelechtendahl *[12-9]*.

After the bubble length exceeds a user-specified minimum length (usually 3-10 cm), the vapor bubble calculation is switched to a different model. Heat flow through the liquid-vapor interfaces is ignored, but axial variations in vapor pressure within the bubble are accounted for. Vaporization of the liquid films in the hot part of the core, and condensation in cooler regions, can lead to high vapor velocities and corresponding appreciable pressure gradients within a vapor bubble. The numerical scheme that is used to calculate the time and space variations in the vapor pressure is an implicit scheme that can handle moderately large (a few milliseconds) time steps. Wave effects tend to be suppressed by the numerical scheme.

Condensation in cooler regions can cause a vapor bubble to shrink. If the bubble size decreased to below the minimum length for the pressure variation model, then the initial uniform pressure model is used again. If the bubble size decreases to less than a minimum bubble size, and if at the same time the rate of decrease of the bubble size exceeds a minimum value, then the bubble is assumed to collapse and the bubble is eliminated from the calculation. When a bubble collapse, the liquid slugs above and below the bubble are combined into one slug, and the flow rate of the combined slug is determined by conserving the combined momentum of the two slugs.

12.5 Uniform Vapor Pressure Model: Small Vapor Bubbles

In this model, the bubble growth is determined by coupling the momentum equations for the liquid slugs, as described in Section 12.2, with an energy balance in the vapor bubble, assuming saturation conditions and spatially uniform pressure and temperature within a bubble. The rate of formation and condensation of vapor is determined by the heat flow through the liquid films on the cladding and structure and through the liquid-vapor interfaces.

Figure 12.5.2 shows the control volume considered in the uniform vapor pressure model. The control volume extends from the lower liquid-vapor interface to the upper one, and from the outer surface of the cladding to the inner surface of the structure. This means that the liquid film remaining on the structure and cladding is included in the control volume. The dots in the figure indicate the locations of the fixed axial mesh points. The piecewise constant representation of the outer cladding radius is consistent with the ability of the code to handle variable flow area.

The primary focus of this model is to obtain the temperature within the vapor bubble. Once the temperature is known, it can be used to calculate the vapor pressure, since saturation conditions have been assumed. The vapor pressure is the driving force for motion of the liquid slugs, so finding the vapor pressures in all bubbles provides the link between conditions in the liquid slugs and conditions in the bubbles and therefore leads to a complete description of conditions throughout the channel.

The vapor temperature is found by taking an energy balance within the control volume described above. The energy balance can be stated qualitatively as

> Net energy transferred to the volume = Net change in energy within the volume.
>
> The energy transferred to the control volume can be divided into two sources:
>
> > Energy transferred = (Heat flow from the cladding and the structure)
> > + (Heat flow through the vapor-liquid interfaces).

Also, the net change in energy is divided between temperature change and phase change.

> Change in energy = (Energy change due to temperature change of vapor present at time t) + (energy change due to vaporization or condensation during the time step).

The remainder of this section will formulate equations that model each portion of the energy balance and that, when combined, will reduce to a set of linear equations that give the vapor temperatures of the bubbles in the channel in terms of known quantities. Section 12.5.1 will discuss heat flow from the cladding and structure, Section 12.5.2 will analyze heat flow through the liquid-vapor interface, Section 12.5.3 will detail the change in vapor energy due to temperature and phase changes, and Section 12.5.4 will produce the final energy balance and the equations governing the bubble temperatures. Section 12.5.5 will then discuss the solution of the bubble temperature equations.

12.5.1 Heat Flow to Vapor from Cladding and Structure

The total heat energy added to vapor bubble K in a time step is

$$E_t = \int_t^{t+\Delta t} \widetilde{Q}\,(\tau)\,d\tau \tag{12.5-1}$$

where $\widetilde{Q}$ is the total heat energy flow rate into the vapor:

$$\widetilde{Q} = \int_s q ds \tag{12.5-2}$$

q is the surface heat flux, and s is the vapor surface. The total heat energy is the sum of the energy flow from the cladding and structure, E_{es}, and the energy flow through the liquid-vapor interfaces, E_i:

$$E_t = E_{es} + E_i \tag{12.5-3}$$

The cladding and structure term, E_{es} is approximated by

$$E_{es} = \frac{\Delta t}{2}\left[Q_{es}\,(K,t) + Q_{es}(K,t+\Delta t)\right] \tag{12.5-4}$$

where Q_{es} is the heat flow from the cladding and structure,

$$Q_{es}\,(K,t) = P_e \int_{z_i(L=1,t,K)}^{z_2(L=2,t,K)} \left[q_e\,(z,t) + \gamma_2 q_s(z,t)\right] dz \tag{12.5-5}$$

with

$z_i\,(2,t,K) =$ height of upper liquid-vapor interface

$z_i\,(1,t,K) =$ height of lower liquid-vapor interface

$P_e =$ perimeter of cladding $= 2\pi r_e$

$r_e =$ nominal radius of cladding

$\gamma_2 =$ ratio of surface area of structure to surface area of cladding

$q_e =$ cladding-to-vapor heat flux

and

$q_s =$ structure-to-vapor heat flux.

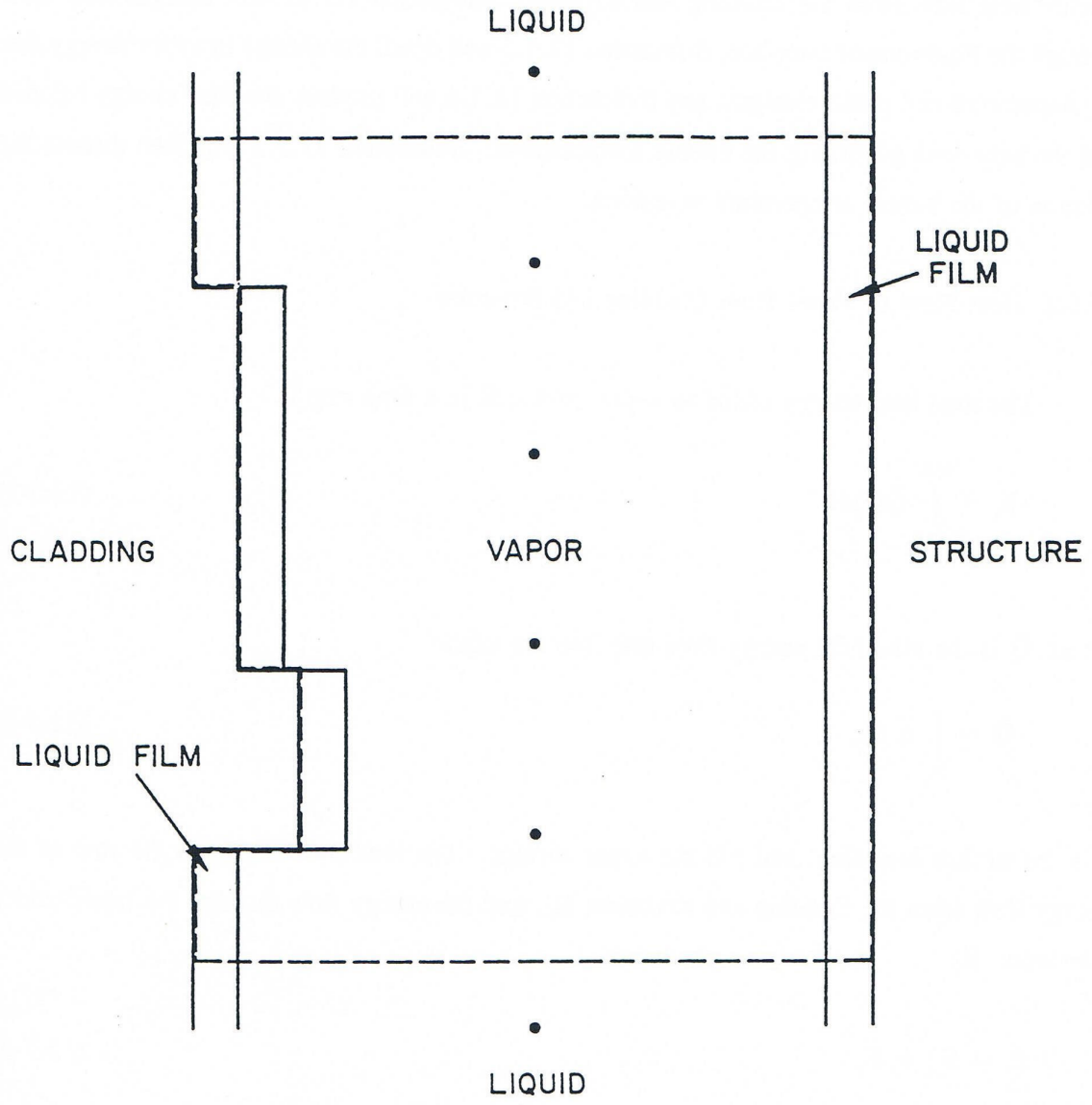


Figure 12.5.1: Control volume for the Uniform Vapor Pressure Model (control volume is outlines by the dashed line)

The heat fluxes are calculated from Newton's law of cooling as

$$q_e\left(z,t\right)=\frac{T_e\left(z,t\right)-T(K,t)}{R_{ec}(z,t)} \tag{12.5-6}$$

and

$$q_s\left(z,t\right)=\frac{T_s\left(z,t\right)-T(K,t)}{R_{sc}(z,t)} \tag{12.5-7}$$

where $T\left(K,t\right)$ is the uniform temperatures within bubble K at time *t* and R_{ec} and R_{sc} are the thermal resistances between the cladding and vapor and he structure and vapor, respectively.

The form of the thermal resistances is best discussed in terms of a thermal network. Focusing first on the cladding-vapor resistance, Figure 12.5.2 shows the thermal network formed by the cladding, liquid film on the cladding, and vapor and gives the thermal resistances associated with each region. Because of the high thermal conductivity of liquid metals, the resistance of the liquid sodium film can be taken to be just the conductive resistance. The total thermal resistance is then the sum of the individual resistances, or

$$R_{ec}=\frac{1}{h_{ec}}+R_{ehf} \tag{12.5-8}$$

where R_{ehf} is the resistance of the cladding, as discussed in Chapter 3 (see Section 3.5.1, Eq. (3.5-7)) and h_{ec} is the heat-transfer coefficient which models the combined resistances of the liquid film and the vapor. The coefficient h_{ec} is just $\frac{1}{\frac{1}{h_c}+\frac{w_{fe}}{k_l}}$; however, this is not as simple an expression as it appears, since the convective heat-transfer coefficient h_c is not a constant or a simple function of temperature. To circumvent this difficulty, a temperature correlation based on physical data was developed for h_{ec} and used in SASSYS-1. This correlation is based on the temperature difference between the cladding and the vapor and is structured as follows. If the cladding is more than 100 K hotter than the vapor, the liquid film is assumed to be at the same temperature as the vapor, which amounts to neglecting the thermal resistance of the vapor. The coefficient h_{ec} then takes the form

$$h_{ec}\left(z\right)=\frac{k_l(z)}{w_{fe}(z)}\text{ if }T_e\left(z\right)>T\left(K\right)+100 \tag{12.5-9}$$

where

$k_l\left(z\right)=$ thermal conductivity of liquid sodium at temperature $T\left(K\right)$

and

$w_{fe}\left(z\right)=$ thickness of liquid film on the cladding

and $T\left(K\right)$ represents an extrapolated vapor temperature for advanced time values of h_{ec}. If the cladding is more than 100 K colder than the vapor, the liquid film is assumed to be at the same temperature as the cladding, and so the resistance of the film is neglected. The heat-transfer coefficient from the vapor to the film is a condensation coefficient h_{cond} which is supplied by the user through the input variable **HCOND**. A reasonable value for **HCOND** is around 6×10^4 W/m^2 K. The coefficient h_{ec} then becomes

$$h_{ec}\left(z\right)=h_{cond}\text{ if }T_e\left(z\right)<T\left(K\right)-100 \tag{12.5-10}$$

For $T\left(K\right)-100\leq T_e\left(z\right)\leq T\left(K\right)+100$,$h_{ec}$ is calculated as an interpolation of the condensation coefficient and the conductive film resistance according to the formula

$$h_{ec}\left(z\right)=h_{cond}+\frac{\frac{k_l(z)}{w_{fe}(z)}-h_{cond}}{1+\exp\left[\frac{T(K)-T_e(z)}{2}\right]},T\left(K\right)-100\leq T_e\left(z\right)\leq T\left(K\right)+100 \tag{12.5-11}$$

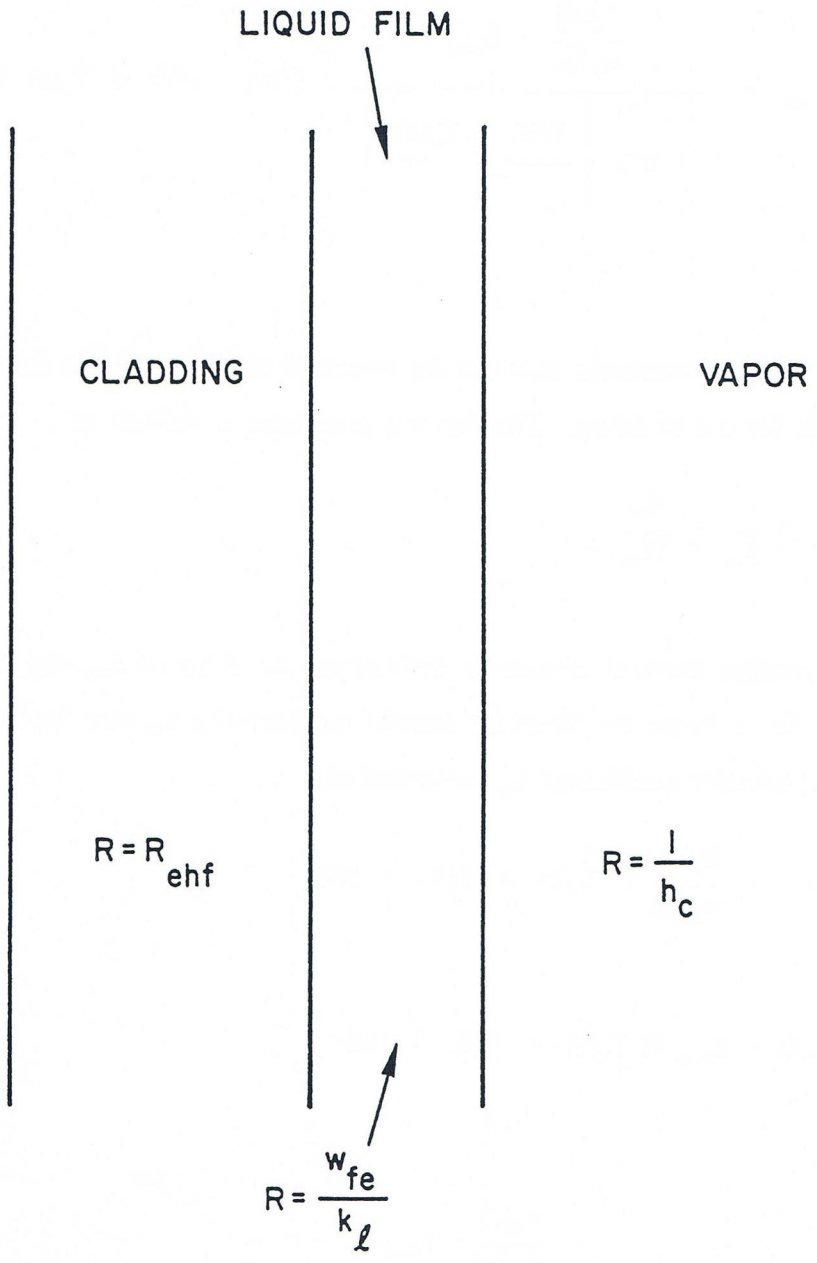


Figure 12.5.2: Thermal Network Formed by the Cladding, Cladding Film, and Vapor

The thermal resistance between the structure and he vapor is calculated using the same procedure as for the cladding. The thermal resistance is defined as

$$R_{sc} = \frac{1}{h_{sc}} + \frac{d_{sti}}{2k_{sti}} \tag{12.5-12}$$

with the structure thermal resistance defined as the ratio of d_{sti}, the thickness of the inner structure node, to twice the structure thermal conductivity k_{sti} (see Section 3.5.2, Eq. (3.5-18)), and the heat-transfer coefficient h_{sc} computed as

$$h_{sc}(z) = \frac{k_l(z)}{w_{fs}(z)} \text{ if } T_s(z) > T(K) - 100 \tag{12.5-13}$$

$$h_{sc}(z) = h_{cond} \text{ if } T_s(z) < T(K) - 100 \tag{12.5-14}$$

and

$$h_{sc}(z) = h_{cond} + \frac{\frac{k_l(z)}{w_{fe}(z)} - h_{cond}}{1 + \exp\left[\frac{T(K) - T_e(z)}{2}\right]}, T(K) - 100 \le T_s(z) \le T(K) + 100 \tag{12.5-15}$$

where $w_{fs}(z)$ is the thickness of liquid film on the structure.

With the cladding-to-vapor and structure-to-vapor heat fluxes now defined, Eq. (12.5-5) can be solved for $Q_{es}(K,t)$ in terms of known quantities. The problem how is to use Eq. (12.5-5) to express $Q_{es}(K, t+\Delta t)$ as a linear function of the change in vapor temperatures over the time step Δt. The difficulty comes from the fact that the advanced time interface positions z_i (which are the limits of integration in the expression for $Q_{es}(K, t+\Delta t)$ are dependent on the advanced time vapor pressures and, therefore, on the advanced time temperatures. However, $Q_{es}(K, t+\Delta t)$ can be written as a linear function of the temperature changes if the interface positions are written as the sum of two functions, one which contains only known quantities and one which is a linear function of the change in vapor temperature. To this end, the interface position at $(t+\Delta t)$ can be written as a linear function

$$z_i(L, t+\Delta t, K) = z_i(L, t, K) + \Delta z_i(K, L) \tag{12.5-16}$$

with the change in position Δz_i given by

$$\Delta z_i(K, L) = \frac{\Delta t}{2}(v_i(L, t, K) + v_i(L, t+\Delta t, K)) \tag{12.5-17}$$

The advanced time interface velocity can also be expressed as a linear function

$$v_i(L, t+\Delta t, K) = v_i(L, t, K) + \Delta v_i(K, L) \tag{12.5-18}$$

with the change in interface velocity computed from the change in liquid slug mass flow rate

$$\Delta v_i(K, L) = \frac{\Delta W(K')}{(\rho_l A_c)_{K',L}} \tag{12.5-19}$$

where

$$K' = K \text{ if } L = 1 \tag{12.5-20}$$

$$K' = K + 1 \text{ if } L = 2 \tag{12.5-21}$$

and $(\rho_l A_c)_{K',L}$ is the product of the liquid density and channel flow area at the bubble interface. It is assumed in Eq. (12.5-19) that the change in interface velocity can be expressed as the change in the average slug velocity

(see Eq. (12.3-1)), with the correction for film thickness ignored. If now the expression for the change in liquid slug mass flow rate, Eq. (12.2-38) derived in Section 3.2, is combined with Eq. (12.5-19) and inserted into Eq. (12.5-8), the advanced time interface velocity becomes

$$v_i\left(L,t+\Delta t,K\right)=v_i\left(L,t,K\right)+\frac{1}{\left(\rho_l A_c\right)_{K',L}}\frac{\Delta t[AA_0\left(K'\right)+\left(\Delta p_{K'-1}-\Delta p_{K'}\right)\theta_2]}{I_1\left(K'\right)+\theta_2\Delta I_1\left(K'\right)+\theta_2 BB_0\left(K'\right)\Delta t} \tag{12.5-22}$$

where Δp_K is the change in pressure in bubble K during the time step and K' is defined as above. Substituting Eq. (12.5-17) and Eq. (12.5-22) into Eq. (12.5-16) gives the advanced time interface position as

$$z_i\left(L,t+\Delta t,K\right)=z_i\left(L,t,K\right)+v_i\left(L,t,K\right)\Delta t+\frac{\Delta t}{2}\frac{1}{\left(\rho_l A_c\right)_{K',L}}\frac{\Delta t[AA_0\left(K'\right)+\left(\Delta p_{K'-1}-\Delta p_{K'}\right)\theta_2]}{I_1\left(K'\right)+\theta_2\Delta I_1\left(K'\right)+\theta_2 BB_0\left(L'\right)\Delta t} \tag{12.5-23}$$

which can be rewritten as

$$z_i\left(L,t+\Delta t,K\right)=z_{i0}\left(K,L\right)+\Delta z'(K,L) \tag{12.5-24}$$

where $z_{i0}\left(K,L\right)$ is the function

$$z_{i0}\left(K,L\right)=z_i\left(L,t,K\right)+\Delta z_0(K,L) \tag{12.5-25}$$

with

$$\Delta z_{i0}\left(K,L\right)=v_i\left(L,t,K\right)+\frac{1}{2}\frac{\Delta W_0\left(K'\right)\Delta t}{\left(\rho_l A_c\right)_{K',L}} \tag{12.5-26}$$

and

$$\Delta W_0\left(K'\right)=\frac{AA_0\left(K'\right)\Delta t}{I_1\left(K'\right)+\theta_2\Delta I_1\left(K'\right)+BB_0\left(K'\right)\theta_2\Delta t} \tag{12.5-27}$$

and $\Delta z'\left(K,L\right)$ is

$$\Delta z'\left(K,L\right)=\frac{dz_i\left(K,L\right)}{dp}(\Delta p_{K'-1}-\Delta p_{K'}) \tag{12.5-28}$$

with

$$\frac{dz_i(K,L)}{dp}=\frac{\theta_2\Delta t^2}{2(\rho_l A_c)_{K',L}[I_1\left(K'\right)+\theta_2\Delta I_1\left(K'\right)+BB_0\left(K'\right)\theta_2\Delta t]} \tag{12.5-29}$$

The expression for $\frac{dz_i}{dp}$ is obtained by taking the derivative with respect to pressure of Eq. (12.5-17), using Eq. (12.5-22). The function z_{i0} is a function only of known quantities and is the approximate position to which the interface would move at $t+\Delta t$ if the bubble pressures did not change over the time step. The additional change in interface position due to bubble pressure changes is accounted for by $\Delta z'$, which is a linear function of the pressure changes.

Using Eq. (12.5-24) for the interface position, the integral for the cladding and structure heat flow $Q_{es}\left(K,t+\Delta t\right)$ can be expressed as the sum of three integrals:

$$\begin{aligned}Q_{es}\left(K,t+\Delta t\right)=&P_e\int_{z_{i0}(K,1)}^{z_{i0}(K,2)}\left[q_e\left(z,t+\Delta t\right)+\gamma_2 q_s\left(z,t\Delta t\right)\right]dz\\&+P_e\int_{z_{i0}(K,2)}^{z_{i0}(K,2)+\Delta z'(K,1)}\left[q_e\left(z,t+\Delta t\right)+\gamma_2 q_s\left(z,t\Delta t\right)\right]dz\\&+P_e\int_{z_{i0}(K,1)+\Delta z'(K,1)}^{z_{i0}(K,1)}\left[q_e\left(z,t+\Delta t\right)+\gamma_2 q_s(z,t\Delta t)\right]dz\end{aligned} \tag{12.5-30}$$

If the vapor temperature $T(K, t+\Delta t)$ is linearized to be $T(K,t)+\Delta T(K)$, the first integral is a function only of $\Delta T(K)$ and known quantities, since the advanced time cladding and structure temperatures are determined by extrapolation from values calculated in the transient heat-transfer module at the previous two heat-transfer time steps and, therefore, are considered known. Using Eq. (3.5-6) for the heat fluxes, the first integral can be written as the sum $I_{e1}(K) + I_{e2}(K)\,\Delta T(K)$, where

$$I_{e1}(K) = P_e \int_{z_{i0}(K,1)}^{z_{i0}(K,2)} \left\{ \frac{T_e(z, t+\Delta t) - T(K,t)}{R_{ec}(z, t+\Delta t)} + \frac{\gamma_2 [T_s(z, t+\Delta t) - T(K,t)]}{R_{sc}(z, t+\Delta t)} \right\} dz \tag{12.5-31}$$

and

$$I_{e2}(K) = -P_e \int_{z_{i0}(K,1)}^{z_{i0}(K,2)} \left[\frac{1}{R_{ec}(z, t+\Delta t)} + \frac{\gamma_2}{R_{sc}(z, t+\Delta t} \right] dz \tag{12.5-32}$$

The second and third integrals are evaluated by assuming that the heat fluxes are constant in space over the domain of integration; this is a reasonable assumption, since Δz is a small quantity. If second-order terms proportional to $\Delta T \Delta z'$ are dropped, the second integral is just $I_{e3}(K)\,\Delta z'(K,2)$, with

$$I_{e3}(K) = P_e \left\{ \frac{T_e[z_{i0}(K,2), t+\Delta t] - T(K,t)}{R_{ec}[z_{i0}(K,2), t+\Delta t]} + \gamma_2 \left[\frac{T_s[z_{i_0}(K,2), t+\Delta t] - T(K,t)}{R_{sc}[z_{i0}(K,2), t+\Delta t]} \right] \right\} \tag{12.5-33}$$

and the third integral is $I_{e3}(K)\,\Delta z'(K,1)$, where

$$I_{e4}(K) = -P_e \left\{ \frac{T_e[z_{i0}(K,1), t+\Delta t] - T(K,t)}{R_{ec}[z_{i0}(K,1), t+\Delta t]} + \gamma_2 \left[\frac{T_s[z_{i_0}(K,1), t+\Delta t] - T(K,t)}{R_{sc}[z_{i_0}(K,1), t+\Delta t]} \right] \right\} \tag{12.5-34}$$

This gives for $Q_{es}(K, t+\Delta t)$

$$Q_{es}(K, t+\Delta t) = I_{e1}(K) + I_{e2}(K)\,\Delta T(K) + I_{e3}(K)\,\Delta z'(K,2) + I_{e4}(K)\,\Delta z'(K,1) \tag{12.5-35}$$

which, by the definition of $\Delta z'$ and the assumption of saturation conditions, is a function only of known quantities and the changes in bubble temperatures.

Note that Eq. (12.5-35) is valid regardless of the direction of motion of either interface. The sign of $\Delta z'$ will be positive for upward interface motion and negative for downward motion, giving the correct sign to the last two terms in Eq. (12.5-35).

The energy flow in E_{es} is therefore given by the linear equation

$$E_{es} = \frac{\Delta t}{2} \left[Q_{es}(K,t) + I_{e1}(K) + \left(I_{e2}(K) + I_{e3}(K) \frac{dz_i(K,2)}{dp} - I_{e4}(K) \frac{dz_i(K,1)}{dp} \right) \Delta p_K - I_{e3}(K) \frac{dz_i(K,1)}{dp} \Delta p_{K+1} + I_{e4}(K) \frac{dz_i(K,1)}{dp} \Delta p_{K-1} \right] \tag{12.5-36}$$

where the definition of Δz^i in terms of Δp has been used.

12.5.2 Heat Flow through Liquid-Vapor Interface

The SASSYS-1 calculation of the heat flow through the liquid-vapor interface is based on the work by Cronenberg *[12-7]*. In the method, the total heat flow through the liquid-vapor interfaces is the sum of an upper interface term I_{iu} and a lower interface term I_{il}

$$E_i = \Delta t(I_{iu} + I_{il}) \tag{12.5-37}$$

where

$$I_{ix} = k_l A_{cx} \left.\overline{\frac{\partial T_{lx}}{\partial \xi}}\right|_{\xi=0} \tag{12.5-38}$$

with

$x = u$ or l

A_{cx} = area of coolant channel

T_{lx} = liquid temperature near interface

k_l = liquid thermal conductivity near interface

ξ = axial distance from interface

$\xi = z - z_i$ for upper interface

$\xi = -(z - z_i)$ for lower interface

and

$\overline{\frac{\partial T_{lx}}{\partial \xi}}$ is the time average of the spatial derivative for the time step

An expression for the coolant temperature derivative $\overline{\frac{\partial T_{lx}}{\partial \xi}}$ can be derived from the general heat conduction equation written for the liquid slug as

$$\alpha \frac{\partial^2 T_l(\xi, t')}{\partial \xi^2} + \frac{Q(\xi, t')}{\rho_l C_l} = \frac{\partial^2 T_l(\xi, t')}{\partial t'} \tag{12.5-39}$$

where

$\alpha = k_l/\rho_l C_l$, the thermal diffusivity of liquid sodium

k_l = thermal conductivity of the liquid

C_l = liquid heat capacity

ρ_l = liquid density

Q = heat input per unit volume in the liquid

T_l = temperature in liquid slug

ξ = distance into liquid slug form liquid-vapor interface

and

t' = time since the vapor bubble started to form

The heat input Q includes both the direct heating Q_c and the heat flow $Q_{ec}+Q_{sc}$ from the cladding and structure to the liquid slug:

$$Q = Q_c + Q_{ec} + Q_{sc} \tag{12.5-40}$$

where the functional form of Q_c is given in Chapter 3, Eq. (3.3-6), and Q_{ec} and Q_{sc} are calculated form Newton's law of cooling in the form of Eq. (3.3-7) and Eq. (3.3-8) in Chapter 3.

The boundary conditions for the problem are

$$T_l\left(\xi = 0, t'\right) = T\left(t'\right), \text{ the vapor temperature at the liquid-vapor interface} \tag{12.5-41}$$

and

$$T_l\left(\xi = \infty, t'\right) < \infty \tag{12.5-42}$$

and the initial condition is

$$T_l(\xi, t' = 0) \tag{12.5-43}$$

The heat conduction equation, together with the initial and boundary conditions, can be solved for T_l through the use of the Laplace transform method. The details of this approach are presented in Section 12.16; here, just the main points will be listed. The procedure involves four main stages:

1) Take the Laplace transform of the heat conduction equation, Eq. (12.5-39). This produces a second-order ordinary differential equation in space.
2) Solve the differential equation produced in stage 1 for the Laplace transform of T_l as a function of ξ.
3) Take the spatial derivative of the Laplace transform of T_l and evaluate it at $\xi = 0$.
4) Take the inverse Laplace transform of the derivative from stage 3, producing $\left.\frac{\partial T_l}{\partial \xi}\right|_{\xi=0}$

This procedure results in the following expression for $\left.\frac{\partial T_l}{\partial \xi}\right|_{\xi=0}$

$$\begin{aligned}\frac{\partial T_l\left(\xi = 0, t'\right)}{\partial \xi} = \frac{-1}{\sqrt{\pi\alpha}} \int_0^{t'} \frac{\partial T_l\left(\xi = 0, \tau\right)}{\partial \tau} \frac{1}{\sqrt{t' - \tau}} d\tau \\ - \frac{T_l\left(\xi = 0, t' = 0\right)}{\sqrt{\alpha \pi t'}} + \int_0^{\infty} d\xi \int_0^{t'} \frac{\xi Q\left(\xi, \tau\right) \exp\left[-\frac{\xi^2}{4\alpha(t' - \tau)}\right]}{2k_l \sqrt{\pi\alpha\left(t' - \tau\right)^3}} d\tau \\ + \int_0^{\infty} \frac{T_l\left(\xi, t' = 0\right) \xi \exp\left[-\frac{\xi^2}{4\alpha t'}\right]}{2\alpha \sqrt{\pi\alpha\left(t'\right)^3}} d\xi\end{aligned} \tag{12.5-44}$$

To evaluate the terms in Eq. (12.5-44), first note that the exponential function in the third and fourth terms decays very rapidly with increasing ξ due to the small thermal diffusivity of liquid sodium. Therefore, for purposes of solving Eq. (12.5-44), it is valid to assume that the initial temperature distribution $(t' = 0)$ is uniform near the interface, or

$$T_l\left(\xi, t' = 0\right) = T_0 \tag{12.5-45}$$

Then the fourth term in Eq. (12.5-44) is easily integrated to give

$$\int_0^{\infty} \frac{T_0}{\alpha} \frac{\xi exp[-\xi^2/4\alpha t']}{2\sqrt{\pi\alpha\left(t'\right)^3}} d\xi = \frac{T_0}{\sqrt{\pi\alpha t'}} \tag{12.5-46}$$

so the second and fourth terms cancel.

Similarly, $Q(\xi,\tau)$ can be approximated as the spatially constant function $Q_{Ox}(\tau)$ near the interface so the third term becomes

$$\int_0^{t'} d\tau \int_0^{\infty} d\xi \frac{\xi Q_{Ox} \exp\left[-\frac{\xi^2}{4\alpha(t'-\tau)}\right]}{2k_l\sqrt{\pi\alpha\left(t'-\tau\right)^3}} = \int_0^{t'} \frac{d\tau Q_{Ox}(\tau)}{k_l}\left(\frac{\alpha}{\pi(t'-\tau}\right)^{1/2} \tag{12.5-47}$$

with $Q_{Ox}(\tau)$ defined as

$$Q_{Ox}(\tau) = \gamma\left[\frac{T_{e_x}(\tau) - T(\tau)}{R_{ecx}} + \gamma_2 \frac{T_{s_x}(\tau) - T(\tau)}{R_{scx}}\right] \tag{12.5-48}$$

where

γ = ratio of cladding surface area to coolant volume

γ_2 = ratio of surface area of structure to surface area of cladding

R_{ecx} = thermal resistance between the cladding and the coolant at the interface

R_{scx} = thermal resistance between the structure and the coolant at the interface

and

$$x = \begin{cases} \text{U at the upper interface} \\ l \text{ at the lower interface} \end{cases}$$

Thus, Eq. (12.5-44) has the reduced form

$$\frac{\partial T_1}{\partial \xi}(\xi = 0, t') = \frac{1}{\sqrt{\pi}} \int_0^{t'} \left[Q_{Ox}(\tau)\frac{\sqrt{a}}{k_l} - \frac{1}{\sqrt{a}}\frac{\partial T_l}{\partial \tau}(\xi = 0, \tau)\right] \frac{d\tau}{\sqrt{t'-r}} \tag{12.5-49}$$

which can be rewritten in terms of *t*, the time since initiation of the transient, through the variable transformation $t' = t - t_b$:

$$\left.\frac{\partial T_{lx}(t)}{\partial \xi}\right|_{\xi=0} = \frac{1}{\sqrt{\pi}} \int_{t_b}^{t} \left[Q_{\mathrm{Ox}}(\tau)\frac{\sqrt{a}}{k_l} - \frac{1}{\sqrt{a}}\frac{\partial T(\tau)}{\partial \tau}\right] \frac{d\tau}{\sqrt{t-r}} \tag{12.5-50}$$

The boundary condition at $\xi = 0$ has been incorporated into this expression. Eq. (12.5-50) will be somewhat easier to work with if the function

$$f_x(\tau) \equiv Q_{\mathrm{Ox}}(\tau)\frac{\sqrt{a}}{k_l} - \frac{1}{\sqrt{a}}\frac{\partial T(\tau)}{\partial \tau} \tag{12.5-51}$$

is used to give the simpler equation

$$\left.\frac{\partial T_{lx}(t)}{\partial \xi}\right|_{\xi=0} = \frac{1}{\sqrt{\pi}} \int_{t_b}^{t} \frac{d\tau}{\sqrt{t-\tau}} f_x(\tau) \tag{12.5-52}$$

Now, the time average over the time step Δt of $\left.\frac{\partial T_{lx}(t)}{\partial \xi}\right|_{\xi=0}$ needed to compute the interface heat flow from Eq. (12.5-60) can be written as

$$\left.\frac{\partial \overline{T}_{l_x}}{\partial \xi}\right|_{\xi=0} = \frac{1}{\Delta t} \int_0^{\Delta t} d\varepsilon \left.\frac{\partial T_{lx}(t_k + \varepsilon)}{\partial \xi}\right|_{\xi=0} \tag{12.5-53}$$

with $\left.\frac{\partial T_{lx}(t)}{\partial \xi}\right|_{\xi=0}$ expressed as in Eq. (12.5-52) and $t = t_k$ the time at the beginning of the time step. To accomplish the goal of expressing the energy balance for the bubble as a linear function of the vapor temperatures,

the integral in Eq. (12.5-53) must be written as a linear function of the vapor temperatures. This in turn means that the function $f_x(\tau)$ must be approximated so that Eq. (12.5-52) can be integrated and so that the resulting expression is linear in the vapor temperature. This can be done by writing $\left.\frac{\partial T_{lx}(t)}{\partial \xi}\right|_{\xi=0}$

$$\begin{aligned}\frac{\partial T_{lx}(t+\varepsilon)}{\partial \xi}|_{\xi=0} &= \frac{1}{\sqrt{\pi}}\int_{t_b}^{t_k+\varepsilon}\frac{d\tau}{\sqrt{t_k-\tau+\varepsilon}}f_x(\tau)\\ = \frac{1}{\sqrt{\pi}}\int_{t_b}^{t_k}\frac{d\tau}{\sqrt{t_k-\tau+\varepsilon}}f_x(\tau) &+ \frac{1}{\sqrt{\pi}}\int_{t_k}^{t_k+\varepsilon}\frac{d\tau}{\sqrt{t_k-\tau+\varepsilon}}f_x(\tau)\end{aligned} \tag{12.5-54}$$

The first term in Eq. (12.5-54) is a function only of known quantities, since the range of integration extends only up to the beginning of the current time step. Therefore, the dependence on advanced time vapor temperatures is contained entirely in the second term of Eq. (12.5-54). To integrate this term, define the quantity $\overline{f}_{x\Delta t}$ to be the simple average of $f_x(\tau)$ over the range of integration:

$$\begin{aligned}\overline{f}_{x\Delta t} &= \frac{1}{\Delta t}\int_{t_k}^{t_k+\Delta t} dt f_x(t)\\ &= \frac{1}{\Delta t}\int_{t_k}^{t_k+\Delta t} dt\left[Q_{\text{Ox}}(t)\frac{\sqrt{a}}{k_l}-\frac{1}{\sqrt{a}}\frac{\partial T(t)}{\partial t}\right]\end{aligned} \tag{12.5-55}$$

If the cladding, structure, and vapor temperatures in the expression given above for Q_{Ox} are assumed linear in time over Δt, $\overline{f}_{x\Delta t}$becomes

$$\begin{aligned}\overline{f}_{x\Delta t} &= \frac{1}{\Delta t}\left[\frac{\sqrt{\alpha}}{k_l}\frac{\Delta t}{2}\left(Q_{\text{Ox}}(t_k)+Q_{\text{Ox}}(t_k+\Delta t)\right)-\frac{1}{\sqrt{\alpha}}\left(T(t_k+\Delta t)-T(t_k)\right)\right]\\ &= \frac{1}{2}\left[Q_{\text{Ox}}(t_k)+Q_{\text{Ox}}(t_k+\Delta t)\right]\frac{\sqrt{\alpha}}{k_l}-\frac{1}{\sqrt{\alpha}}\frac{T(t_k+\Delta t)-T(t_k)}{\Delta t}\end{aligned} \tag{12.5-56}$$

Since Δt (and therefore ε) is small, it is reasonable to approximate $f_x(t)$ as $\overline{f}_{x\Delta t}$ from t_k to $t_k+\varepsilon$. Thus, the second term in Eq. (12.5-54) becomes

$$\frac{1}{\sqrt{\pi}}\int_{t_k}^{t_k+\varepsilon}\frac{d\tau}{\sqrt{t_k-\tau+\varepsilon}}f_x(\tau) = \frac{1}{\sqrt{\pi}}\int_{t_k}^{t_k+\varepsilon}\frac{d\tau}{\sqrt{t_k-\tau+\varepsilon}}\overline{f}_{x\Delta t} \tag{12.5-57}$$

This equation can be integrated easily by making the change of variable $\tau' = t_k-\tau+\varepsilon$ to give

$$\frac{1}{\sqrt{\pi}}\int_{t_k}^{t_k+\varepsilon}\frac{d\tau}{\sqrt{t_k-\tau+\varepsilon}}\overline{f}_{x\Delta t} = \overline{f}_{x\Delta t}\int_0^{\epsilon}\frac{d\tau'}{\sqrt{\tau'}} = 2\sqrt{\varepsilon}\overline{f}_{x\Delta t} \tag{12.5-58}$$

which is linear in $T(T_k+\Delta t)$.

The interface heat flow has now been expressed as a linear function of the advanced time vapor temperature. However, one problem remains; namely, evaluation of the integral in the first term in Eq. (3.5-47). One simple way to handle this is to approximate $f_x(\tau)$ by some average value $\overline{f}_x$ over the range of integration, giving

$$\begin{aligned}\int_{t_b}^{t_k}\frac{d\tau}{\sqrt{t_k+\varepsilon-\tau}}f_x(\tau) &= \overline{f}_x\int_{t_b}^{t_k}\frac{d\tau'}{\sqrt{\tau_k-\varepsilon-\tau}}\\ &= \overline{f}_x\int_{\varepsilon}^{t_k+\varepsilon-t_b}\frac{d\tau'}{\sqrt{\tau'}}\\ &= 2\overline{f}_x(\sqrt{\tau_k+\varepsilon-\tau_b}-\sqrt{\varepsilon})\end{aligned} \tag{12.5-59}$$

where the integral has been evaluated using the change of variable $\tau' = t_k+\varepsilon-\tau$ as above. Now, the problem reduces to one of selecting an appropriate value for $\overline{f}_x$. The range of integration is too large to use the simple

average as was done above. An alternative approach, is to examine the integral in the first term of Eq. (12.5-54) and try to develop an expression for $\overline{f}_x$ which is a function of t_k and which can be computed form values at earlier time steps through a recursion relation. To this end, use the fact that ε is small to approximate the first integral in Eq. (12.5-54) as

$$\begin{aligned}\int_{t_b}^{t_k} \frac{d\tau}{\sqrt{t_k+\varepsilon-\tau}} f_x(\tau) &\approx \int_{t_b}^{t_k} \frac{d\tau}{\sqrt{t_k-\tau}} f_x(\tau) \\ &= \int_0^{t_k-t_b} \frac{d\tau'}{\sqrt{\tau'}} f_x(t_k-\tau')\end{aligned} \tag{12.5-60}$$

The transformation $\tau' = t_k - \tau$ was used in the last integral in Eq. (12.5-60). Now, define the function $g_x(\tau')$ as

$$g_x(\tau') = f_x(t_k-\tau') \tag{12.5-61}$$

and the weighted average of g_x as

$$\overline{g}_x(t_k) = \frac{\int_0^{t_k-t_b} \frac{d\tau'}{\sqrt{\tau'}} g_x(\tau')}{\int_0^{t_k-t_b} \frac{d\tau'}{\sqrt{\tau'}}} \tag{12.5-62}$$

As will be shown below, $\overline{g}_x(t_k)$can be determined form a recursion relation.

Rearranging Eq. (12.5-62) gives the integral in Eq. (12.5-60) the form

$$\int_0^{t_k-t_b} \frac{d\tau'}{\sqrt{\tau'}} f_x(t_k-\tau') = 2\sqrt{\tau_k - t_b}\,\overline{g}_x(t_k) \tag{12.5-63}$$

The function $\overline{f}_x$can now be expressed in terms of $\overline{g}_x(t_k)$by combining Eq. (12.5-59), Eq. (12.5-60), and Eq. (12.5-63) to give

$$\begin{aligned}2\overline{f}_x\left(\sqrt{t_k+\varepsilon-\tau_b}-\sqrt{\varepsilon}\right) &= \int_{t_b}^{t_k} \frac{d\tau}{\sqrt{t_k+\varepsilon-\tau}} f_x(\tau) \\ &\approx \int_{t_b}^{t_k} \frac{d\tau}{\sqrt{t_k-\tau}} f_x(\tau) \\ &= 2\sqrt{t_k-t_b}\,\overline{g}_x(t_k)\end{aligned} \tag{12.5-64}$$

Since ε is small compared to $t_k - t_b$ for all but the times right after the bubble has formed, it is reasonable to approximate $\sqrt{\tau_k+\varepsilon-t_b}-\sqrt{\varepsilon}\,\sqrt{t_k-t_b}$and therefore to choose

$$\overline{f}_x = \overline{g}_x(t_k) \tag{12.5-65}$$

as the approximation to $f_x(\tau)$ for $t_b \le \tau \le t_k$This expression can be computed from a recursion relation by writing $\overline{g}_x(t_k+\Delta t)$as

$$\begin{aligned}\overline{g}_x(t_k+\Delta t) &= \int_0^{t_k+\Delta t-t_b} \frac{d\tau'}{\sqrt{\tau'}} f_x(t_k+\Delta t-\tau') \Big/ \int_0^{t_k+\Delta t-t_b} \frac{d\tau'}{\sqrt{\tau'}} \\ = \frac{1}{2\sqrt{t_k+\Delta t-t_b}} &\left[\int_0^{\Delta t} \frac{d\tau'}{\sqrt{\tau'}} f_x(t_k+\Delta t-\tau') + \int_{\Delta t}^{t_k+\Delta t-t_b} \frac{d\tau'}{\sqrt{\tau'}} f_x(t_k+\Delta t-\tau')\right]\end{aligned} \tag{12.5-66}$$

Substituting the approximations $f_x(t_k+\Delta t-\tau') = \overline{f}_{x\Delta t}$ for $0 \le \tau' \le \Delta t$ and $f_x(t_k+\Delta t-\tau') = \overline{g}(t_k)$

for $\Delta t \le \tau' \le t_k + \Delta t - t_b$ as used above reduces the integrals in Eq. (12.5-66) to

$$\begin{aligned}\overline{g}_x\left(t_k+\Delta t\right) &= \frac{1}{2\sqrt{t_k+\Delta t-t_b}}\left[\int_0^{\Delta t}\frac{d\tau'}{\sqrt{\tau'}}\overline{f}_{x\Delta t}+\int_{\Delta t}^{t_k+\Delta t-t_b}\frac{d\tau'}{\sqrt{\tau'}}\overline{g}_x\left(t_k\right)\right]\\ &= \frac{1}{\sqrt{t_k+\Delta t-t_b}}\left[\overline{f}_{x\Delta t}\sqrt{\Delta t}+\overline{g}_x\left(t_k\right)\left(\sqrt{t_k+\Delta t-t_b}-\sqrt{\Delta t}\right)\right]\\ &= \overline{g}_x\left(t_k\right)\left[1-\frac{\sqrt{\Delta t}}{\sqrt{t_k+\Delta t-t_b}}\right]+\overline{f}_{x\Delta t}\frac{\sqrt{\Delta t}}{\sqrt{t_k+\Delta t-t_b}}\end{aligned} \tag{12.5-67}$$

so that once the advanced time vapor temperatures are computed, the value of $\overline{g}_x$for the next time step can be calculated.

With a means now available for obtaining , $\overline{g}_x(t_x)$the expression for $\left.\frac{\partial T_{lx}}{\partial\xi}\right|_{\xi=0}$ in Eq. (12.5-54) can be considered fully determined, and Eq. (12.5-54) can be written

$$\left.\frac{\partial T_{lx}(t_k+\varepsilon)}{\partial\xi}\right|_{\xi=0} = \frac{1}{\sqrt{\pi}}\left[2\overline{g}_x\left(t_k\right)\left(\sqrt{t_k+\varepsilon-t_b}-\sqrt{\varepsilon}\right)+2\overline{f}_{x\Delta t}\sqrt{\varepsilon}\right] \tag{12.5-68}$$

The time average of Eq. (12.5-68) is then

$$\begin{aligned}\left.\overline{\frac{\partial T_{lx}}{\partial\xi}}\right|_{\xi=0} &= \frac{2}{\sqrt{\pi}}\frac{1}{\Delta t}\int_0^{\Delta t}d\varepsilon\left[\overline{g}_x\left(t_k\right)\left(\sqrt{t_k+\varepsilon-t_b}-\sqrt{\varepsilon}\right)+\overline{f}_{x\Delta t}\sqrt{\varepsilon}\right]\\ &= \frac{4}{3\sqrt{\pi}}\frac{1}{\Delta t}\left[\overline{g}_x\left(t_k\right)\left(t_k+\Delta t-t_b\right)^{\frac{3}{2}}-\left(t_k-t_b\right)^{\frac{3}{2}}-\left(\Delta t\right)^{\frac{3}{2}}+\overline{f}_{x\Delta t}\left(\Delta t\right)^{\frac{3}{2}}\right]\end{aligned} \tag{12.5-69}$$

Thus, the interface heat flow I_{ix} is, from Eq. (12.5-38) and substituting for $\overline{f}_{x\Delta t}$

$$\begin{aligned}I_{ix} = k_l A_{cx}\frac{4}{3\sqrt{\pi}}\frac{1}{\Delta t}\Bigg[&\overline{g}_x\left(t\right)\left(\left(t+\Delta t-t_b\right)^{\frac{3}{2}}-\left(t-t_b\right)^{\frac{3}{2}}-\left(\Delta t\right)^{\frac{3}{2}}\right)\\ &+\left(\Delta t\right)^{\frac{3}{2}}\left(\frac{1}{2}\left[Q_{Ox}\left(t\right)+Q_{Ox}\left(t+\Delta t\right)\right]\frac{\sqrt{\alpha}}{k_l}-\frac{1}{\sqrt{\alpha}}\frac{T\left(t+\Delta t\right)-T(t)}{\Delta t}\right)\Bigg]\end{aligned} \tag{12.5-70}$$

which, substituting the definition of Q_{Ox}, is

$$\begin{aligned}I_{ix} = \frac{k_l}{\sqrt{\pi}}A_{cx}\frac{4}{3}\Bigg[&\overline{g}_x\left(t\right)\left(\frac{\left(t+\Delta t-t_b\right)^{\frac{3}{2}}-\left(t-t_b\right)^{\frac{3}{2}}}{\Delta t}-\sqrt{\Delta t}\right)\\ &+\sqrt{\Delta t}\Bigg(\frac{\gamma}{2}\left[\frac{T_{ex}\left(t\right)+T_{ex}\left(t+\Delta t\right)-T\left(t\right)-T\left(t+\Delta t\right)}{R_{ecx}}\right]\\ &+\gamma_2\frac{T_{sx}\left(t\right)+T_{sx}\left(t+\Delta t\right)-T\left(t\right)-T\left(t+\Delta t\right)}{R_{scx}}\Bigg]\frac{\sqrt{\alpha}}{k_l}\\ &-\frac{1}{\sqrt{\alpha}}\frac{T\left(t+\Delta t\right)-T\left(t\right)}{\Delta t}\Bigg]\Bigg)\\ = \frac{4}{3}\frac{k_l}{\sqrt{\pi}}A_{cx}\overline{g}_x\left(t\right)&\left(\frac{\left(t+\Delta t-t_b\right)^{\frac{3}{2}}-\left(t-t_b\right)^{\frac{3}{2}}}{\Delta t}-\sqrt{\Delta t}\right)\\ &+\frac{2}{3}\frac{\sqrt{\alpha}}{\sqrt{\pi}}\gamma\sqrt{\Delta t}A_{cx}\Bigg[\frac{T_{ex}\left(t\right)+T_{ex}\left(t+\Delta t\right)-2T\left(t\right)}{R_{ecx}}\\ &+\gamma_2\frac{T_{sx}\left(t\right)+T_{sx}\left(t+\Delta t\right)-2T\left(t\right)}{R_{scx}}\Bigg]\\ -\Delta T\Bigg[&\frac{2}{3}\frac{\sqrt{\alpha}}{\sqrt{\pi}}A_{cx}\gamma\sqrt{\Delta t}\left(\frac{1}{R_{ecx}}+\frac{\gamma_2}{R_{scx}}\right)+\frac{4}{3}\frac{k_l}{\sqrt{\pi}}\frac{A_{ex}}{\sqrt{\alpha}}\frac{1}{\sqrt{\Delta t}}\Bigg]\end{aligned} \tag{12.5-71}$$

where $T(t+\Delta t) = T(t) + \Delta T$. Using this expression in Eq. (12.5-37) for E_i, the total heat flow through the liquid-vapor interfaces, gives

$$E_i = (I_{i0} + I_{i1}\Delta T)\,\Delta t \tag{12.5-72}$$

where

$$\begin{aligned} I_{i0} = \frac{k_l}{\sqrt{\pi}} \Bigg\{ \sum_{x=u,l} A_{cx} \Bigg[\frac{2}{3}\frac{\sqrt{\alpha}}{k_l}\gamma\sqrt{\Delta t} \Bigg[\frac{T_{ex}(t) + T_{ex}(t+\Delta t) - 2T(t)}{R_{ecx}} \\ + \gamma_2 \frac{T_{sx}(t) + T_{sx}(t+\Delta t) - 2T(t)}{R_{scx}} \Bigg] \\ + \frac{4}{3}\overline{g}_x(t) \left[\frac{(t+\Delta t - t_b)^{\frac{3}{2}} - (t-t_b)^{\frac{3}{2}} - \sqrt{\Delta t}}{\Delta t} \right] \Bigg] \Bigg\} \end{aligned} \tag{12.5-73}$$

and

$$I_{ix} = \frac{k_l}{\sqrt{\pi}}\frac{4}{3}\sqrt{\Delta t} \left\{ -\frac{1}{\Delta t\sqrt{\alpha}}(A_{cu} + A_{cl}) - \frac{\gamma}{2k_l}\sqrt{\alpha} \left[\sum_{x=u,l} A_{cx} \left(\frac{1}{R_{ecx}} + \frac{\gamma_2}{R_{scx}} \right) \right] \right\} \tag{12.5-74}$$

This completes the task of expressing the heat flow through the interface as a linear function of the advanced time vapor temperatures.

12.5.3 Change in Vapor Energy

The heat flow into the bubble control volume is used both to produce new vapor and to raise the temperature of already existing vapor. During a time interval Δt, the vapor temperature goes from T to $T + \Delta T$, the pressure goes from p_v to $p_v + \Delta p_v$, the density goes from ρ_v to $\rho_v + \Delta\rho_v$, the bubble volume goes from V_v to $V_v + \Delta V_v$, and the vapor energy changes by ΔE. The changes Δp and $\Delta\rho_v$ are related to ΔT by the requirement that saturation conditions prevail in the vapor.

Two processes contribute to the energy change ΔE. One is the heating of the quantity of vapor present at the beginning of the time step from temperature T to temperature $T + \Delta T$. The other is the vaporizing of some of the liquid film to form additional vapor, giving a total vapor mass of $(\rho_v + \Delta\rho_v)(V_v + \Delta V_v)$ at the end of the time step. However, it is not straightforward to formulate an expression for the energy change by directly considering the heating of the vapor (because the volume and density changes which take place during the heating) and the vaporization of some of the liquid film (because the amount of film vaporized is unknown). Therefore a thermodynamically equivalent path is followed which does allow straightforward expression of the energy change. This path can be described in the following three steps:

Step 1: Condense the vapor in the bubble at time *t* to liquid at constant pressure and temperature:

$$\Delta E(1) = -(\rho_v V_v)\,\lambda \tag{12.5-75}$$

where λ is the heat of vaporization at time *t* and

$$\begin{aligned} V_v &= \int A_c dz \\ &= A_c(\text{JST})\,\Delta z'(\text{JST}) + \sum_{\text{JC=JST}+1}^{\text{JEND}-1} A_c(\text{JC})\Delta z(\text{JC}) + A_c(\text{JEND})\Delta z'(\text{JEND}) \end{aligned} \tag{12.5-76}$$

Refer to Figure 12.2.1 for the notation.

Step 2: Heat the liquid from Step 1 to $T + \Delta T$:

$$\Delta E\,(2) = C_l\,(\rho_v V_v)\,\Delta T \tag{12.5-77}$$

where C_l is the heat capacity of the liquid and the compressibility of the liquid is neglected.

Step 3: Vaporize the liquid from Step 2 plus enough liquid from the film to fill the volume $V_v + \Delta V$:

$$\Delta E\,(3) = (\rho_v + \Delta\rho_v)(V_v + \Delta V)(\lambda + \Delta\lambda) \tag{12.5-78}$$

If the vapor undergoes a net energy loss rather than a gain, the liquid in Step 2 shows a temperature drop of ΔT and part of the vapor in Step 3 condenses onto the cladding and/or structure.

The energy change is then

$$\Delta E = \Delta E\,(1) + \Delta E\,(2) + \Delta E(3) \tag{12.5-79}$$

or, neglecting second-order terms,

$$\Delta E = \rho_v V_v \Delta\lambda + \rho_v \lambda \Delta V + \lambda V_v \Delta\rho_v + \rho_v V_v C_l \Delta T \tag{12.5-80}$$

Now, the energy change must be expressed as a linear function of the change in vapor temperature ΔT. To do this, first look at the volume change ΔV. This term is currently modeled as the change in volume at the liquid-vapor interfaces due to interface motion, neglecting any volume change due to flow area changes caused by cladding motion during the time step. Accordingly, using earlier notation, ΔV for bubble K is just

$$\begin{aligned}\Delta V = [A_{cl}\,(t+\Delta t)\,z_i\,(1, t+\Delta t, K) - A_{cl}\,(t)\,z_i\,(1, t, K)] + [A_{cu}\,(t+\Delta t)\,z_i\,(2, t+\Delta t, K)\\ - A_{cu}\,(t)\,z_i\,(2, t, K)]\end{aligned} \tag{12.5-81}$$

The flow area A_c at the interfaces is written as a time-dependent function to account for the possibility that an interface might cross from one mesh segment to another during the time step. Since the flow area can vary from mesh segment to mesh segment, this might result in a change in interface flow area from t to Δt.

To simplify the expression for ΔV, define an average interface area at the lower interface as

$$\overline{A}_c\,(K, 1) = \frac{\int_{z_i(1,t,K)}^{z_i(1,t+\Delta t,K)} A_c\,(z)\,dz}{z_i\,(1, t+\Delta t, K) - z_i(1, t, K)} \tag{12.5-82}$$

A similar definition can be made at the upper interface. The ΔV becomes

$$\begin{aligned}\Delta V = \overline{A}_c\,(K, 2)\,[z_i\,(2, t+\Delta t, K) - z_i\,(2, t, K)]\\ -\overline{A}_c(K, 1)[z_i\,(1, t+\Delta t, K) - z_i(1, t, K)]\end{aligned} \tag{12.5-83}$$

The advanced time interface positions $z_i\,(L, t+\Delta t, K)$ can be expressed in terms of the changes in pressure at the interfaces via Eq. (12.5-24) to give

$$\begin{aligned}\Delta V &= \Delta V_0 + \overline{A}_c\,(K, 2)\,\Delta z'\,(K, 2) - \overline{A}_c\,(K, 1)\,\Delta z'\,(K, 1)\\ &= \Delta V_0 + \overline{A}_c\,(K, 2)\,\frac{dz_i\,(K, 2)}{dp}\,(\Delta p_K - \Delta p_{K+1})\\ &\quad -\overline{A}_c\,(K, 1)\,\frac{dz_i\,(K, 1)}{dp}(\Delta p_{K-1} - \Delta p_K)\end{aligned} \tag{12.5-84}$$

where

$$\Delta V_0 = \overline{A}_c\,(K, 2)\,\Delta z_0\,(K, 2) - \overline{A}_c\,(K, 1)\,\Delta z_0(K, 1) \tag{12.5-85}$$

Using Eq. (12.5-84) in Eq. (12.5-80) for the energy change ΔE produces an expression for ΔE in terms of the changes in $\lambda, \rho_v,$ and p_v as well as T. To reduce this to a linear equation in ΔT, the changes in $\lambda, \rho_v,$ and p_v are approximated by

$$\Delta\lambda = \Delta T \frac{d\lambda}{dT} \tag{12.5-86}$$

$$\Delta\rho_v = \Delta T \frac{d\rho_v}{dT} \tag{12.5-87}$$

and

$$\Delta p = \Delta T \frac{dp_v}{dT} \tag{12.5-88}$$

where the temperature derivatives are evaluated along the saturation curve. Incorporating Eq. (12.5-86) into Eq. (12.5-80) results in a formulation for the change in energy within the control volume which is a linear function of ΔT:

$$\Delta E = \Delta E_0 + \Delta E_1 \Delta T\,(K) + \Delta E_2 \Delta T\,(K+1) + \Delta E_3 \Delta T(K-1) \tag{12.5-89}$$

where

$$\Delta E_0 = \lambda \rho_v \Delta V_0 \tag{12.5-90}$$

$$\Delta E_1 = \lambda \left\{ \rho_v \frac{dp\,(K)}{dT} \left[\overline{A}_c\,(K,1) \frac{dz_i\,(K,1)}{dp} + \overline{A}_c\,(K,2) \frac{dz_i\,(K,2)}{dp} \right] + (V_v + \Delta V_0) \frac{d\rho_v}{dT} \right\} + \rho_v \Delta V_0 \frac{d\lambda}{dT} + \rho_v V_v C_l \tag{12.5-91}$$

$$\Delta E_2 = 0 \text{ if } K = K_{vn} = \text{last bubble in channel} \tag{12.5-92}$$

and otherwise,

$$\Delta E_2 = -\rho_v \lambda \overline{A}_c\,(K,2) \frac{dz_i\,(K,2)}{dp} \frac{dp(K+1)}{dT} \tag{12.5-93}$$

$$\Delta E_3 = 0 \text{ if } K = 1 \tag{12.5-94}$$

and otherwise,

$$\Delta E_2 = -\rho_V \lambda \overline{A}_c\,(K,1) \frac{dz_i\,(K,1)}{dp} \frac{dp(K-1)}{dT} \tag{12.5-95}$$

12.5.4 Energy Balance

As discussed at the beginning of this section, an energy balance exists between the energy transferred to the control volume and the change in energy within the volume. The change in energy is given by Eq. (12.5-89), derived in the previous subsection. The energy transferred to the volume, E_t, is the sum of the energy flow from the cladding and structure, E_{es}, and the energy flow through the liquid-vapor interfaces, E_i; this was expressed in Eq. (12.5-3). Substituting Eq. (12.5-36) and Eq. (12.5-72), the expressions for E_{es} and E_i derived in Section 12.5.1 and Section 12.5.2, respectively, into Eq. (12.5-3) gives the total energy transferred to the control volume as

$$E_t = E_{t0} + E_{t1} \Delta T\,(K) + E_{t2} \Delta T\,(K+1) + E_{t3} \Delta T(K-1) \tag{12.5-96}$$

where

$$E_{t0} = \Delta t[\frac{Q_{es}\left(K,t\right) + I_{e1}\left(K\right)}{2} + I_{i0}] \tag{12.5-97}$$

$$E_{t1} = \Delta t\Bigg[\frac{I_{e2}\left(K\right)}{2} + I_{i1} + \frac{I_{e3}\left(K\right)}{2}\frac{dz_i\left(K,2\right)}{dp}\frac{dp\left(K\right)}{dT} - \frac{I_{e4}\left(K\right)}{2}\frac{dz_i\left(K,1\right)}{dp}\frac{dp\left(K\right)}{dT}\Bigg] \tag{12.5-98}$$

$$E_{t2} = 0 \text{ if } K = K_{vn} \tag{12.5-99}$$

and otherwise,

$$E_{t2} = -\frac{\Delta t}{2} I_{e3}\left(K\right)\frac{dz_i\left(K,2\right)}{dp}\frac{dp(K+1)}{dT} \tag{12.5-100}$$

$$E_{t3} = 0 \text{ if } K = 1 \tag{12.5-101}$$

and otherwise,

$$E_{t3} = \frac{\Delta t}{2} I_{e4}\left(K\right)\frac{dz_i(K,1)}{dp}\frac{dp(K-1)}{dT} \tag{12.5-102}$$

The overall energy balance is then

$$E_t = \Delta E \tag{12.5-103}$$

If the expression for E_t from Eq. (12.5-96) and that for ΔE from Eq. (12.5-89) are inserted into Eq. (12.5-103), the result is a linear equation in terms of the changes in the vapor temperatures of bubbles $K-1, K,$ and $K+1$:

$$C_4\left(K\right) + C_1\left(K\right)\Delta T\left(K\right) + C_2\left(K\right)\Delta T\left(K+1\right) + C_3\left(K\right)\Delta T\left(K-1\right) = 0 \tag{12.5-104}$$

where

$$C_4\left(K\right) = \Delta E_0\left(K\right) - E_{t0}\left(K\right) \tag{12.5-105}$$

$$C_1\left(K\right) = \Delta E_1\left(K\right) - E_{t1}(K) \tag{12.5-106}$$

$$C_2\left(K\right) = \Delta E_2\left(K\right) - E_{t2}(K) \tag{12.5-107}$$

$$C_3\left(K\right) = \Delta E_3\left(K\right) - E_{t3}(K) \tag{12.5-108}$$

12.5.5 Vapor Temperatures

Eq. (12.5-104) can be written for each uniform pressure bubble in the channel. The equations are then solved for the changes in vapor temperature for each bubble as follows. First, each bubble in the channel is checked in order from the bottom of the channel to the top to determine whether it is a uniform-pressure or variable-pressure bubble. This determines how the uniform-pressure bubbles are distributed throughout the channel and allows the temperature calculation to be carried out simultaneously for all bubbles that are in any one group of bubbles (e.g., if the lowest bubble in the channel is a large, pressure-gradient bubble with four small constant pressure bubbles above it, the temperatures in the four small bubbles will be computed simultaneously). In general, if a series of N bubbles of uniform vapor pressure extends from bubble K_b to bubble K_t, then the temperatures in the N bubbles are calculated by solving a set of linear equations consisting of Eq. (12.5-104) written for each of

the N bubbles. These N equations will be written in terms of N unknowns if $\Delta T\left(K_b-1\right)$ and $\Delta T\left(K_t+1\right)$ are set to extrapolated values (or to zero, if $K_b=1$ or $K_t=K_{vN}$) and the coefficient C_4 is modified to be

$$C_4\left(K_b\right) \rightarrow C_4\left(K_b\right)+C_3\left(K_b\right)\Delta T\left(K_b-1\right), \text{ if } K_b>1 \tag{12.5-109}$$

$$C_4\left(K_t\right) \rightarrow C_4\left(K_t\right)+C_2\left(K_t\right)\Delta T\left(K_t+1\right), \text{ if } K_t<K_{\text{vN}} \tag{12.5-110}$$

The N equations are then solved using Gaussian elimination. After the bubble temperatures are obtained, the saturation conditions are used to obtain the bubble pressures.

12.6 Vapor Pressure Gradient Model: Large Vapor Bubbles

As mentioned above, whenever a bubble grows to a length greater than a user-specified value (usually 5-50 cm), the vapor bubble calculation is switched from the uniform-vapor-pressure model to a vapor-pressure-gradient model. Saturation conditions are assumed in this model, so the vapor energy equation can be combined with the mass continuity equation. The vapor continuity and momentum equations are solved simultaneously for each node in a bubble using a fully implicit finite differencing scheme.

The equations described in this section are only for the vapor. Any liquid in a bubble region is assumed to be a film on the cladding and structure, and this film is treated separately, as described at the beginning of Section 12.1. The vapor calculation is influenced by the liquid film in only two ways: heat flow through the film provides the vapor source for the vapor calculations; and if a two-phase friction factor multiplier for the friction between the vapor and the film is used, then the film thickness determines the value of this multiplier.

12.6.1 Continuity and Momentum Equations

The vapor continuity equation can be written as

$$\frac{\partial W}{\partial z}+\frac{\partial\left(\rho_v A_c\right)}{\partial t}=\frac{Q}{\lambda}A_C \tag{12.6-1}$$

while the vapor momentum equation is given by

$$\frac{\partial p}{\partial z}=-\frac{1}{A_c}\frac{\partial}{\partial z}\frac{W^2}{\rho_v A_c}-\frac{f_v W|W|}{2\rho_v D_h A_c^2}+F_c-\frac{1}{A_c}\frac{\partial W}{\partial t}-\frac{W|W|}{2\rho_v A_c^2}\frac{\partial k_{or}}{\partial z} \tag{12.6-2}$$

The symbols used in Eq. (12.6-1) and Eq. (12.6-2) are defined as follows:

W = vapor mass flow rate

z = axial height

ρ_v = sodium vapor density

A_c = coolant flow area

t = time

Q = heat flow rate per unit volume

λ = heat of vaporization

p = pressure

f_v = friction factor

D_h = hydraulic diameter

F_c = condensation momentum loss term

k_{or} = orifice pressure drop term

Note that, because the flow area A_c is included in the mass and momentum equations as a function of both space and time, the model is able to treat problems involving variable flow areas.

The heat flow rate per unit volume is given by

$$Q = \gamma \left[\frac{T_e - T}{R_{ec}} + \gamma_2 \frac{T_s - T}{R_{sc}} \right] \tag{12.6-3}$$

where

λ = ratio of cladding surface area to coolant volume

T_e = cladding temperature

T_s = structure temperature

T = vapor temperature

R_{ec} = thermal resistance between the cladding and the vapor

R_{sc} = thermal resistance between the structure and the vapor

γ_2 = ratio of structure surface are to cladding surface area

The thermal resistances R_{ec} and R_{sc} account for the resistance to heat flow of the cladding, liquid film, and vapor and are computed according to the expressions discussed in Section 12.5.1.

The expression for the heat flow rate Q can be rewritten as the sum of the heat flow rate through the cladding, Q_e, and the heat flow rate through the structure, Q_s, where

$$Q_e = \gamma \frac{T_e - T}{R_{ec}} \tag{12.6-4}$$

and

$$Q_s = \gamma\gamma_2 \frac{T_s - T}{R_{sc}} \tag{12.6-5}$$

These two quantities are used to compute the condensation momentum loss term F_c. This term accounts for momentum lost from the system when flowing sodium vapor condenses on cladding or structure that is colder than the vapor. Only a momentum loss term is included, since it is assumed that the system does not gain momentum when vaporization occurs from cladding or structure that is hotter than the sodium film. The condensation momentum loss is simply the product of the rate of mass condensing per unit volume, $\frac{Q}{\lambda}$, times the average velocity, $\frac{W}{2\rho_v A_c}$. Splitting the condensation momentum loss term into separate contributions from the cladding and structure then gives.

$$F_c = F_{ce} + F_{cs} \tag{12.6-6}$$

where

$$F_{ce} = \begin{cases} 0 & Q_e \geq 0, \\ \frac{Q_e W}{2\rho_v A_c \lambda} & Q_e < 0 \end{cases} \tag{12.6-7}$$

and

$$F_{cs} = \begin{cases} 0 & Q_s \geq 0, \\ \frac{Q_s W}{2\rho_v A_c \lambda} & Q_s < 0 \end{cases} \tag{12.6-8}$$

These definitions reflect the assumption that momentum is lost when condensation of vapor occurs (Q_e, Q_s negative) but that no momentum change occurs when liquid is vaporized (Q_e, Q_s positive).

The friction factor, f_v is expressed as

$$f_v = A_{fr}\left(\frac{D_h\,|W|}{\mu A_c}\right)^{b_{fr}} f_{tp} \tag{12.6-9}$$

where μ is the vapor viscosity and A_{fr} and b_{fr} are input constants. The quantity in parentheses, $D_h|W|/\left(\mu A_c\right)$, is just the Reynolds number of the sodium vapor. The two-phase friction factor multiplier, f_{tp}, is based on a correlation by Wallis *[12-10]*:

$$f_{tp} = 1 + 300 f_{2\phi}(w_{fe} + \gamma_2 w_{fs})/[(1+\gamma_2)\,D_h] \tag{12.6-10}$$

where $f_{2\phi}$ is set to 1.0 when the two-phase multiplier is used and is set to 0 by the user if a smooth tube friction factor is described.

Two approximations are made to Eq. (12.6-1) and Eq. (12.6-2) before finite differencing. First, in the continuity equation, the term

$$\frac{\partial(\rho_v A_c)}{\partial t} = \rho_v \frac{\partial A_c}{\partial t} + A_c \frac{\partial \rho_v}{\partial t} \tag{12.6-11}$$

is taken to be

$$\frac{\partial(\rho_v A_c)}{\partial t} = A_c \frac{\partial \rho_v}{\partial t} \tag{12.6-12}$$

So that A_c is assumed constant in time (but not space) over the time step for purposes fo solving the mass and momentum equations for the vapor pressures and mass flow rates. This assumption is also made with the variables D_h, γ, and γ_2. This simplification is a reasonable one, since one of these four quantities changes rapidly with time. Second, in the momentum equations, the momentum convection term can be expanded as

$$\frac{\partial}{\partial z}\left(\frac{W^2}{\rho_v A_c}\right) = \frac{1}{A_c}\frac{\partial}{\partial z}\left(\frac{W^2}{\rho_v}\right) + \frac{W^2}{\rho_v}\frac{\partial}{\partial z}\left(\frac{1}{A_c}\right) \tag{12.6-13}$$

Any changes in area over a spatial node are incorporated into the orifice term, and so $\frac{W^2}{\rho_v}\frac{\partial}{\partial z}(\frac{1}{A_c})$ is included in the $\frac{\partial k_{or}}{\partial z}$ term. Therefore,

$$\frac{\partial}{\partial z}\left(\frac{W^2}{\rho_v A_c}\right) = \frac{1}{A_c}\frac{\partial}{\partial z}\left(\frac{W^2}{\rho_v}\right) \tag{12.6-14}$$

so A_c appears piecewise constant in space. This is also true for D_h, γ, and γ_2.

Therefore, the continuity and momentum equations to be differenced are

$$\frac{\partial W}{\partial z} + A_c \frac{\partial \rho_v}{\partial t} = \frac{Q}{\lambda} A_c \tag{12.6-15}$$

and

$$\frac{\partial p}{\partial z} = -\frac{1}{A_c^2}\frac{\partial}{\partial z}\left(\frac{W^2}{\rho_v}\right) - A_{fr}\frac{W\,|W|^{1+b_{fr}}}{2\rho_v D_h^{1-b_{fr}} A_c^{2+b_{fr}}} + F_c - \frac{1}{A_c}\frac{\partial W}{\partial t} - \frac{W\,|W|}{2\rho_v A_c^2}\frac{\partial k_{or}}{\partial z} \tag{12.6-16}$$

12.6.2 Finite Differencing

The mass and momentum equations must now be reduced to algebraic equations that can be coded into SASSYS-1. This is accomplished in three steps: (1) the equations are discretized using finite differences, (2) the advanced time terms are linearized, and (3) saturation conditions are imposed to limit the independent variables to the change in mass flow rate and the change in pressure. The details of these three steps are presented below. The result is the set of algebraic equations Eq. (12.6-45), Eq. (12.6-79), Eq. (12.6-87) (discretized form of the continuity equation at an interior node, lower interface, and upper interface of a bubble, respectively), Eq. (12.6-132), Eq. (12.6-147), and Eq. (12.6-153) (discretized form of the momentum equation in the interior, the lower interface, and the upper interface). These equations are then solved according to the procedure discussed in Section 12.6.3.

As indicated in Figure 12.6.1, some of the variables in the mass and momentum equations are calculated at node boundaries, whereas others are calculated at node mid-points. In general, midpoint quantities are used as if they were constant over the node. The quantities defined at node boundaries are z, T, p, W, ρ_v, μ, and the velocity V. At liquid-vapor interfaces, these quantities are defined at the interface, and the vapor velocity is set to the value of the interface velocity.

Midpoint quantities include T_e, T_s, R_{ec}, A_c, D_h, γ, γ_2, k_{or}, γ, Q, F_e, and the spatially averaged coolant temperature $\overline{T}$. With all variables, a subscript 1 indicates that the variable is evaluated at the beginning of the time step; a subscript 2 means the variable is taken to be at the end of the time step.

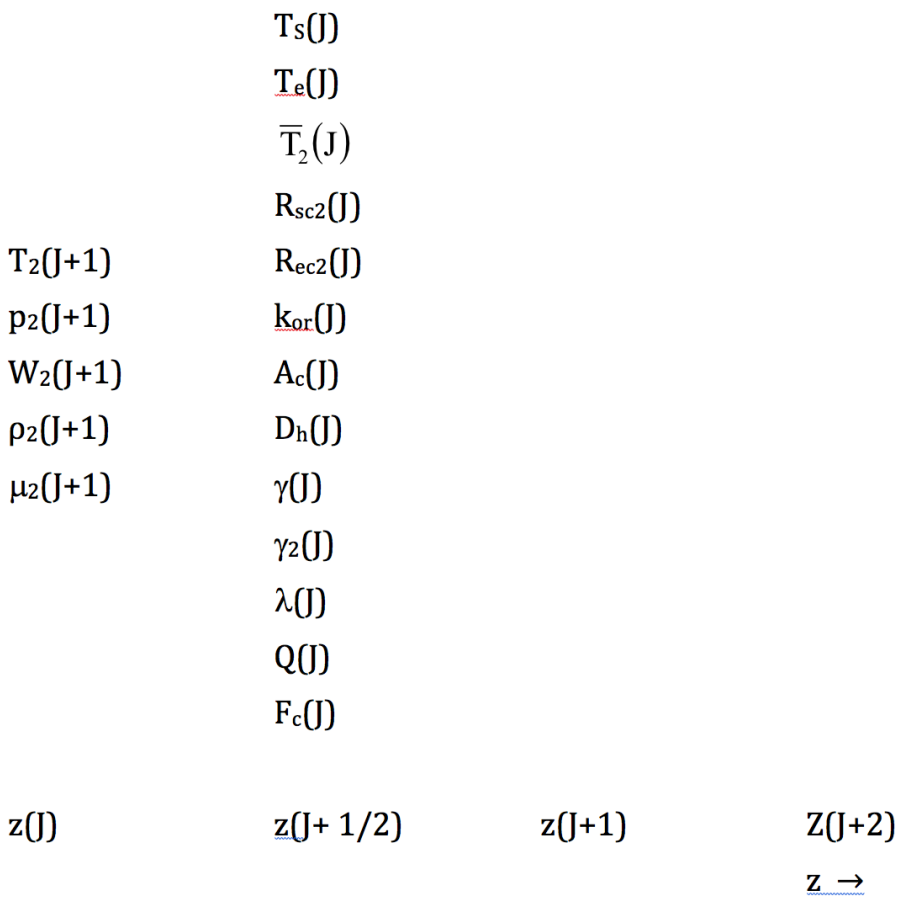


Figure 12.6.1: SASSYS-1 Voiding Model Axial Coolant Mesh Variable Placement

12.6.2.1 Finite Differencing of the Continuity Equation for a Mesh Segment Contained Entirely in One Bubble

This section gives a detailed explanation of how the continuity equation is expressed in finite difference form. The extra factors required to model bubble-liquid interfaces correctly are not included here; they will be discussed in Section 12.6.2.2. Therefore, the equations to be developed in this section apply as they stand only to axial mesh segments which are entirely contained within the bubble.

12.6.2.1.1 Differencing of the Continuity Equation

The terms making up the continuity equation are differenced as follows:

$$\left.\frac{\partial W}{\partial z}\right|_{J+1/2} = \frac{W_2(J+1) - W_2(J)}{z(J+1) - z(J)} \tag{12.6-17}$$

and

$$A_c \left.\frac{\partial \rho_v}{\partial t}\right|_{J+1/2} = A_c(J)\frac{\rho_2(J) - \rho_1(J) + \rho_2(J+1) - \rho_1(J+1)}{2\Delta t} \tag{12.6-18}$$

These differencing schemes are illustrated in Figure 12.6.2. Also,

$$\begin{aligned} Q_2(J) &= \gamma(J)\left[\frac{T_{e_2}(J) - \overline{T}_2(J)}{R_{ec_2}} + \gamma_2(J)\frac{T_{s_2}(J) - \overline{T}_2(J)}{R_{sc_2}(J)}\right] \\ &= Q_{e_2}(J) + Q_{s_2}(J) \end{aligned} \tag{12.6-19}$$

Therefore, the differenced continuity equation becomes

$$\begin{aligned} \frac{W_2(J+1) - W_2(J)}{z(J+1) - z(J)} + A_c(J)\frac{\rho_2(J) - \rho_1(J) + \rho_2(J+1) - \rho_1(J+1)}{2\Delta t} \\ = A_c(J)\frac{Q_2(J)}{\lambda_2(J)} \end{aligned} \tag{12.6-20}$$

12.6.2.1.2 Linearization of Advanced Time Terms

Eq. (12.6-20) can be linearized by expressing the advanced time terms in the form

$$x_2(J) = x_1(J) + \Delta x(J) \tag{12.6-21}$$

where *x* is any quantity. This produces

$$\rho_2(J) = \rho_1(J) + \Delta\rho(J) \tag{12.6-22}$$

$$\lambda_2(J) = \lambda_1(J) + \Delta\lambda(J) \tag{12.6-23}$$

$$W_2(J) = W_1(J) + \Delta W(J) \tag{12.6-24}$$

$$T_2(J) = T_1(J) + \Delta T(J) \tag{12.6-25}$$

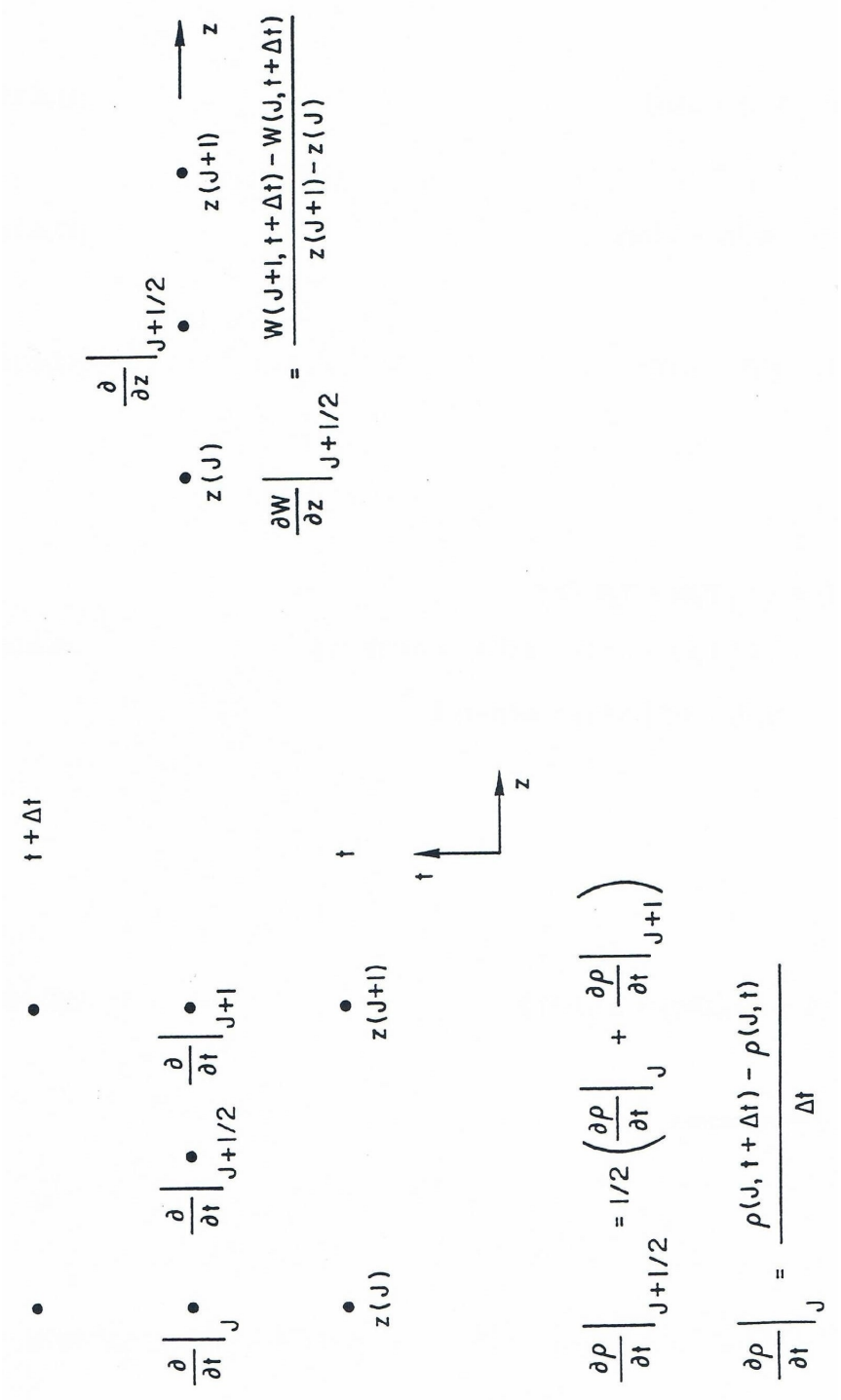


Figure 12.6.2: Evaluation of Derivatives at an Interior Node in the SASSYS-1 Voiding Model

Since

$$\begin{aligned}\overline{T}_2(J) &= \frac{1}{2}\left(T_2(J) + T_2(J+1)\right) \\ &= \frac{1}{2}\left(T_1(J) + \Delta T(J) + T_1(J+1) + \Delta T(J+1)\right) \\ &= \overline{T}_1(J) + \frac{1}{2}(\Delta T(J) + \Delta T(J+1))\end{aligned} \tag{12.6-26}$$

then

$$\Delta\overline{T}(J) = \frac{1}{2}\left(\Delta T(J) + \Delta T(J+1)\right) \tag{12.6-27}$$

The heat flux then becomes

$$\begin{aligned}Q_2(J) = \gamma(J)\Bigg[&\frac{T_{e_2}(J) - \overline{T}_1(J) - \Delta\overline{T}(J)}{R_{ec_2}(J)} + \gamma_2(J)\frac{T_{s_2}(J) - \overline{T}_1(J) - \Delta\overline{T}(J)}{R_{sc_2}(J)}\Bigg] \\ = \gamma(J)\Bigg[&\frac{T_{e_2}(J) - \overline{T}_1(J)}{R_{ec_2}(J)} + \gamma_2(J)\frac{T_{s_2}(J) - \overline{T}_1(J)}{R_{sc_2}(J)} \\ &- \frac{(\Delta T(J) + \Delta T(J+1))}{2}\left[\frac{1}{R_{ec_2}(J)} + \frac{\gamma_2(J)}{R_{sc_2}(J)}\right]\Bigg]\end{aligned} \tag{12.6-28}$$

The heat flux expression can be written in a more compact form by defining the variable $Q_o(J)$

$$Q_0(J) = \gamma(J)\left[\frac{T_{e_2}(J) - \overline{T}(J)}{R_{ec_2}(J)} + \gamma_2(J)\frac{T_{s_2}(J) - \overline{T}_1(J)}{R_{sc_2}(J)}\right] \tag{12.6-29}$$

The derivative with respect to vapor temperature of Q_o is then

$$\frac{dQ_0}{dT_J} = -\gamma(J)\left[\frac{1}{R_{ec_2}(J)} + \frac{\gamma_2(J)}{R_{sc_2}(J)}\right] \tag{12.6-30}$$

These can be substituted into the above equation for $Q_2(J)$ to give

$$Q_2(J) = Q_0(J) + \frac{1}{2}\frac{dQ_0}{dT_J}\left(\Delta T(J) + \Delta T(J+1)\right) \tag{12.6-31}$$

Substituting Eq. (12.6-22) through Eq. (12.6-25) and Eq. (12.6-31) for the advanced time terms in the differenced continuity equation, Eq. (12.6-20) gives

$$\begin{aligned}&\frac{W_1(J+1) - W_1(J) + \Delta W(J+1) - \Delta W(J)}{z(J+1) - z(J)} + A_c(J)\frac{\Delta\rho(J) + \Delta\rho(J+1)}{2\Delta t} \\ &= \frac{A_c(J)}{\lambda_1(J) + \Delta\lambda(J)}\left(Q_0(J) + \frac{1}{2}\frac{dQ_0}{dT_J}\left(\Delta T(J) + \Delta T(J+1)\right)\right)\end{aligned} \tag{12.6-32}$$

If $\Delta\lambda(J) \ll \lambda_1(J)$,

$$\frac{1}{\lambda_1(J) + \Delta\lambda(J)} \approx \frac{1}{\lambda_1(J)}\left[1 - \frac{\Delta\lambda(J)}{\lambda_1(J)}\right] \tag{12.6-33}$$

which changes Eq. (12.6-32) to

$$\begin{aligned}&\frac{W_1(J+1) - W_1(J) + \Delta W(J+1) - \Delta W(J)}{z(J+1) - z(J)} + A_c(J)\frac{\Delta\rho(J) + \Delta\rho(J+1)}{2\Delta t} \\ &= A_c(J)\frac{Q_0(J)}{\lambda_1(J)}\left(1 + \frac{\Delta T(J) + \Delta T(J+1)}{2Q_0(J)}\frac{dQ_0}{dT_J}\right)\left(1 - \frac{\Delta\lambda(J)}{\lambda_1(J)}\right)\end{aligned} \tag{12.6-34}$$

Eliminating second-order terms gives the linearized differenced continuity equation

$$\frac{W_1(J+1)-W_1(J)+\Delta W(J+1)-\Delta W(J)}{z(J+1)-z(J)}+A_c(J)\frac{\Delta\rho(J)+\Delta\rho(J+1)}{2\Delta t} = A_c(J)\frac{Q_0(J)}{\lambda_1(J)}\left(1+\frac{\Delta t(J)+\Delta T(J+1)}{2Q_0(J)}\frac{dQ_0}{dT_J}-\frac{\Delta\lambda(J)}{\lambda_1(J)}\right) \tag{12.6-35}$$

12.6.2.1.3 Utilization of Saturation Conditions to Limit the Independent Variables

Since saturation conditions are assumed, the independent variables can be limited to changes in flow rate and pressure, ΔW and Δp. ρ, T, and λ are all functions of p only. Therefore, the following difference terms can be expressed in terms of Δp:

$$\Delta T(J)=\Delta p(J)\frac{dT}{dp_J} \tag{12.6-36}$$

$$\Delta\rho(J)=\Delta p(J)\frac{d\rho}{dp_J} \tag{12.6-37}$$

$$\Delta\lambda(J)=\frac{1}{2}\left(\Delta p(J)+\Delta p(J+1)\right)\frac{d\lambda}{dp_J} \tag{12.6-38}$$

with $\Delta p(J)=p_2(J)-p_1(J)$.

All derivatives are evaluated at the start of the time step. When Eq. (12.6-36) through Eq. (12.6-38) are inserted into Eq. (12.6-35), the result is a differenced continuity equation which is linear in ΔW and Δp:

$$\frac{W_1(J+1)-W_1(J)+\Delta W(J+1)-\Delta W(J)}{z(J+1)-z(J)}+A_c(J)\frac{1}{2\Delta t}\left(\Delta p(J)\frac{d\rho}{dp_J}+\Delta p(J+1)\frac{d\rho}{dp}(J+1)\right) = A_c(J)\frac{Q_0(J)}{\lambda_1(J)}\left[1+\left(\Delta p(J)\frac{dT}{dp_J}+\Delta p(J+1)\frac{dT}{dp_{J+1}}\right)\frac{1}{2Q_0(J)}\frac{dQ_0}{dT_J} -\left(\Delta p(J)+\Delta p(J+1)\right)\frac{1}{2\lambda_1(J)}\frac{d\lambda}{dp_J}\right] \tag{12.6-39}$$

Multiply through by $(z(J+1)-z(J))\,\Delta t$ and define the coefficients

$$c_{1,j}=\left(z(J+1)-z(J)\right)A_c(J)\left[\frac{1}{2}\frac{d\rho}{dp_J}-\frac{\Delta t}{2\lambda_1(J)}\frac{dT}{dp_J}\frac{dQ_0}{dT_J}+\frac{Q_0(J)\Delta t}{2\lambda_1^2(J)}\frac{d\lambda}{dp_J}\right] \tag{12.6-40}$$

$$c_{2,j}=-\Delta t \tag{12.6-41}$$

$$c_{3,j}=\left(z(J+1)-z(J)\right)A_c(J)\left[\frac{1}{2}\frac{d\rho}{dp_{J+1}}-\frac{\Delta t}{2\lambda_1(J)}\frac{dT}{dp_{J+1}}\frac{dQ_0}{dT_J}+\frac{Q_0(J)\Delta t}{2\lambda_1^2(J)}\frac{d\lambda}{dp_J}\right] \tag{12.6-42}$$

$$c_{4,J}=\Delta t \tag{12.6-43}$$

Also, define the source term

$$h_J=W_1(J)-W_1(J+1)+\left(z(J+1)-z(J)\right)A_c(J)\frac{Q_0(J)}{\lambda_1(J)}\Delta t \tag{12.6-44}$$

Then the differenced mass equation becomes

$$c_{1,J}\Delta p(J)+c_{2,J}\Delta W(J)+c_{3,J}\Delta p(J+1)+c_{4,J}\Delta W(J+1)=h_j \tag{12.6-45}$$

12.6.2.2 Finite Differencing of the Continuity Equation for a Mesh Segment Which Contains a Bubble Interface

12.6.2.2.1 Adjustment of the Form of the Continuity Equation to Account for Interface Motion

The representation of the partial derivative with respect to time is not as straightforward at an interface as it is at an interior node. Referring to Figure 12.6.3, in which node boundary J is the interface, the partial derivative of the density ρ is still calculated as

$$\left.\frac{\partial \rho}{\partial t}\right|_{J+1/2} = \frac{1}{2}\left[\left.\frac{\partial \rho}{\partial t}\right|_{J} + \left.\frac{\partial \rho}{\partial t}\right|_{J+1}\right] \tag{12.6-46}$$

The partial derivatives at J and $J+1$ are computed just as they are for an interior node; in particular,

$$\left.\frac{\partial \rho}{\partial t}\right|_{J} = \frac{\rho\left(z\left(J,t\right),t+\Delta t\right) - \rho\left(z\left(J,t\right),t\right)}{\Delta t} \tag{12.6-47}$$

so the partial derivative at J is differenced as though the interface were stationary at $z\left(J,t\right)$. However, the interface is moving at a velocity v^i and moves a distance $v^i \Delta t$ from $z\left(J,t\right)$ to $z\left(J,t+\Delta t\right)$ during the time step, so the spatial derivative in the continuity equation must be differenced from $z\left(J+1,t+\Delta t\right)$ to $z\left(J,t+\Delta t\right)$, rather than to $z\left(J,t\right)$. Thus, the continuity equation for an interface node will involve advanced time quantities at three spatial points ($z\left(J,t\right)$, $z\left(J,t+\Delta t\right)$, and $z\left(J+1,t+\Delta t\right)$) rather than at two. This difficulty can be resolved if the partial derivative with respect to time is expressed in terms of $\rho\left(z\left(J,t+\Delta t\right),t+\Delta t\right)$ rather than $\rho\left(z\left(J,t\right),t+\Delta t\right)$ by introducing the Lagrangian total derivative, $\frac{d\rho}{dt}$.

$$\frac{d\rho}{dt} = \frac{\partial \rho}{\partial t} + v\frac{\partial \rho}{\partial z} \tag{12.6-48}$$

or

$$\frac{\partial \rho}{\partial t} = \frac{d\rho}{dt} - v\frac{\partial \rho}{\partial z} \tag{12.6-49}$$

with

$$\frac{d\rho}{dt} = \frac{\rho\left(z\left(J,t+\Delta t\right),t+\Delta t\right) - \rho\left(z\left(J,t\right),t\right) + \rho\left(z(J+1),t+\Delta t\right) - \rho(z(J+1),t)}{2\Delta t} \tag{12.6-50}$$

$$v = \frac{v^i}{2} \tag{12.6-51}$$

and

$$\frac{\partial \rho}{\partial z} = \frac{\rho\left(z\left(J,t+\Delta t\right),t+\Delta t\right) - \rho(z(J+1),t+\Delta t)}{z\left(J,t+\Delta t\right) - z\left(J+1,t+\Delta t\right)} \tag{12.6-52}$$

The factor of (1/2) appears in the velocity term to represent the average of the velocity of the interface point J (which is v^i) and that of the point $J+1$ (which is zero). This formulation is equivalent to expressing $\rho\left(z\left(J,t\right),t+\Delta t\right)$ as an interpolated value between $\rho\left(z\left(J,t+\Delta t\right)\right)$ and $\rho\left(z(J+1),t+\Delta t\right)$. The continuity equation now takes the form

$$\frac{\partial W}{\partial z} + A_c\frac{d\rho}{dt} - v\frac{\partial \rho}{\partial z} = A_c\frac{Q}{\lambda} \tag{12.6-53}$$

Let a node boundary quantity x be represented at the lower interface by $x^i_j(J1)$ and at the upper interface by $x^i_j(J2)$ where $j = 1$ or 2 to designate the beginning or end of the time step. A node midpoint quantity y is

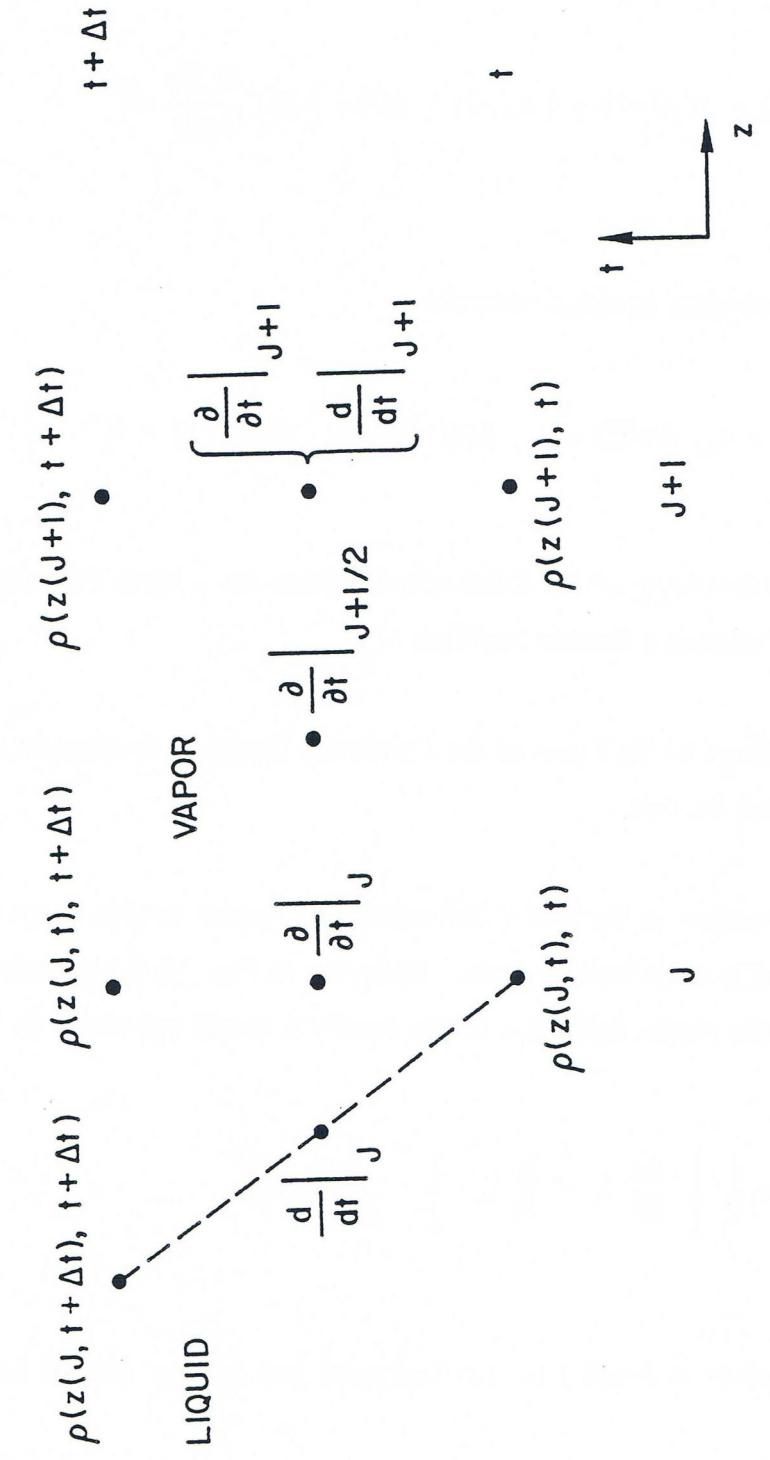


Figure 12.6.3: Evaluation of the total Derivative at a Liquid-Vapor Interface

represented at the lower interface by $y_j(J1)$ (since it is contained in node $J1$) and at the upper interface by $y_j(J2-1)$ (since it is located in node $J2-1$). As shown in Fig. 12.6-4, $J1$ is the number of the fixed mesh node boundary below the lower interface, while $J2$ is the fixed node boundary number above the upper interface. Looking first at the lower interface, the total time derivative of the density is then

$$\frac{d\rho}{dt} = \frac{\rho_2(J1+1) - \rho_1(J1+1) + \rho_2^i(J1) - \rho_1^i(J1)}{2\Delta t} \tag{12.6-54}$$

or, using Eq. (12.6-22) and Eq. (12.6-37) to express $\frac{d\rho}{dt}$ in terms of the change in vapor pressure,

$$\frac{d\rho}{dt} = \frac{1}{2\Delta t}\left(\Delta\rho(J1+1)\frac{d\rho}{dp_{J1+1}} + \Delta\rho^i(J1)\frac{d\rho^i}{dp_{J1}}\right) \tag{12.6-55}$$

The spatial derivative term becomes

$$v\frac{\partial\rho}{\partial z} = \frac{v_2^i(J1)}{2}\frac{(\rho_2(J1+1) - \rho_2^i(J1))}{(z(J1+1) - z_2^i(J1))} \tag{12.6-56}$$

The advanced time velocity is expressed in the linearized form of Eq. (12.6-21) as

$$v_2^i(J1) = v_1^i(J1) + \Delta v^i(J1) \tag{12.6-57}$$

It is apparent from Eq. (12.6-56) and Eq. (12.6-57) that the added complication of treating the bubble interface introduces two extra unknowns beyond those described in Section 12.6.2.1, namely, the change in interface velocity Δv^i and the advanced time interface position z_2^i (the current time interface velocity, v_1^i, is assumed known). As will be shown in Section 12.6.2.2.2 and Section 12.6.2.2.3, both of these quantities can be expressed as functions of Δp and of known terms, so that their effect on the final differenced equation is to add terms to the coefficients $c_{1,J}$ and $c_{3,J}$ and to the source term h_J in Eq. (12.6-45).

12.6.2.2.2 Formulation of Interface Velocity as a Function of Pressure Changes

If the bubble under consideration is labeled bubble K, the change in interface velocity is calculated from the change in mass flow rate $\Delta W_l(K)$ in liquid slug K, which is the slug below bubble K:

$$\Delta v^i(J1) = \frac{\Delta W_l(K)}{\rho_l^i(J1)A_c(J1)} f_{rwt}(K) \tag{12.6-58}$$

where $f_{rwt}(K)$ is the liquid film rewetting factor at the top of slug K and $\rho_l^i(J1)$ is the liquid sodium density at the interface. The film rewetting factor accounts for the effect of liquid film on the cladding and structure on the interface velocity; this has already been discussed in Section 12.3. From Eq. (12.3-6) of Section 12.3, the liquid film rewetting factor is

$$f_{rwt} = \frac{v^i}{v_l} = \frac{1 - P_e\left(w_{fe}v_{fe} + \gamma_2 w_{fs}v_{fs}\right)/(v_l A_c)}{1 - P_e\left(w_{fe} + \gamma_2 w_{fs}\right)/A_c} \tag{12.6-59}$$

The change in mass flow rate is computed from the liquid slug momentum equation given in Eq. (12.2-38):

$$\Delta W_l(K) = \frac{\left(AA_0(K) + \theta_2\left(\Delta p_b(K) - \Delta p_t(K)\right)\right)\Delta t}{I_1(K) + \theta_2\Delta I_1(K) + BB_0(K)\,\theta_2\Delta t} \tag{12.6-60}$$

with $\theta_1 = \theta_2 = 1/2$ for the semi-implicit method and $\theta_1 = 0$, $\theta_2 = 1$ for the fully implicit formulation. The pressure changes are defined as

$\Delta p_b(K)$ = change in pressure at lower end of liquid slug K.

$\Delta p_t(K)$ = change in pressure at upper end of liquid slug K.

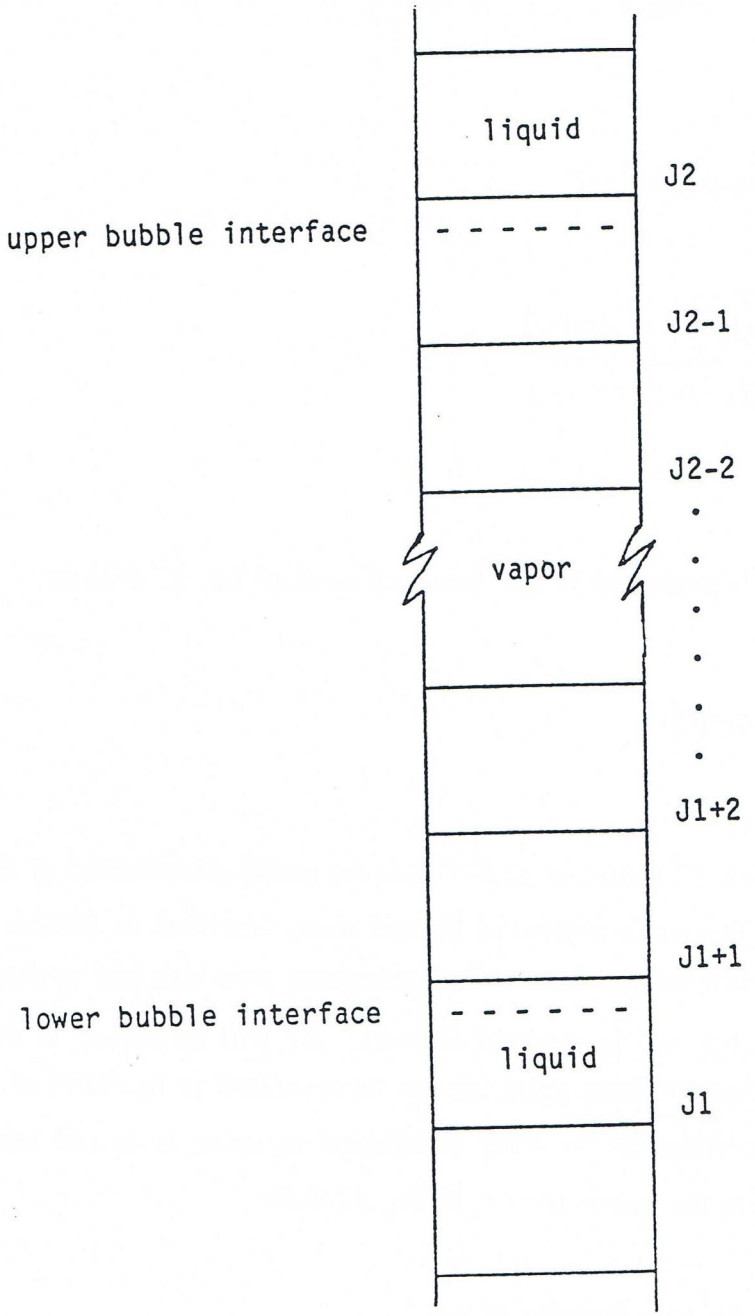


Figure 12.6.4: Placement of a Bubble in the SASSYS-1 Axial Coolant Mesh

Since $\Delta p_t\,(K)$ is the pressure change at the bubble interface,

$$\Delta p_t\,(K) = \Delta p^i(J1) \tag{12.6-61}$$

Therefore, $\Delta v^i(J1)$ can be split into two parts as

$$\Delta v^i(J1) = \Delta v_0^i(J1) + \Delta p^i(J1)\frac{dv^i}{dp_{J1}} \tag{12.6-62}$$

with

$$\Delta v_0^i(J1) = \frac{f_{fwt}(K)}{\rho_l^i(J1)A(J1)}\frac{\left(AA_0\,(K) + \theta_2\Delta p_b\,(K)\right)\Delta t}{I_1\,(K) + \theta_2\Delta I_1\,(K) + BB_0\,(K)\,\theta_2\Delta t} \tag{12.6-63}$$

being the change in interface velocity with no change in interface pressure and

$$\frac{dv^i}{dp_{J1}} = -\frac{f_{rwt}\,(K)}{\rho_l^i(J1)A(J1)}\frac{\theta_2\Delta t}{I_1\,(K) + \theta_2\Delta I_1\,(K) + BB_0\,(K)\,\theta_2\Delta t} \tag{12.6-64}$$

being the velocity change with respect to the change in interface pressure. With the change in interface velocity now expressed as a linear function of the change in interface pressure Eq. (12.6-62), the advanced time interface velocity in the expression for the spatial derivative term $v\frac{\partial\rho}{\partial z}$ Eq. (12.6-56) can be replaced using Eq. (12.6-57) and Eq. (12.6-62). In addition, the advanced time densities in Eq. (12.6-56) can be linearized through Eq. (12.6-22) and written in terms of pressure changes form Eq. (12.6-37). This gives $v\frac{\partial\rho}{\partial z}$ as

$$\begin{aligned} v\frac{\partial\rho}{\partial z} = \frac{1}{2}\left(v_1^i(J1) + \Delta v_0^i(J1) + \Delta p_i(J1)\frac{dv^i}{dp_{J1}}\right) \\ \times\frac{\rho_i(J1+1) + \Delta p(J1+1)\frac{d\rho}{dp_{J1+1}} - \rho_1^i(J1) - \Delta p^i(J1)\frac{d\rho^i}{dp_{J1}}}{z(J1+1) - z_2^i(J1)} \end{aligned} \tag{12.6-65}$$

Now, all that remains is to express the interface position z_2^i as a function of the change in pressure; this will be done in the next subsection.

12.6.2.2.3 Formulation of Interface Position as a Function of Pressure Changes

The interface height can be written as

$$z_2^i(J1) = z_1^i(J1) + \frac{\Delta t}{2}\left(v_1^i(J1) + v_2^i(J1)\right) \tag{12.6-66}$$

which, substituting Eq. (12.6-57) and Eq. (12.6-62) for v_2^i, becomes

$$\begin{aligned} z_2^i(J1) &= z_1^i(J1) + \frac{\Delta t}{2}v_1^i(J1) + \left(v_1^i(J1) + \Delta v_0^i(J) + \Delta p^i(J1)\frac{dv^i}{dp_{J1}}\right) \\ &= z_1^i(J1) + v_1^i(J1)\Delta t + \frac{\Delta t}{2}v_0^i(J1) + \Delta p^i(J1)\frac{\Delta t}{2}\frac{dv^i}{dp_{J1}} \end{aligned} \tag{12.6-67}$$

Define

$$z_0^i(J1) = z_1^i(J1) + v_1^i(J1)\Delta t + \frac{\Delta t}{2}\Delta v_0^i(J1) \tag{12.6-68}$$

as the interface height with no pressure change at the interface and

$$\frac{dz^i}{dp_{J1}} = \frac{1}{2}\Delta t\frac{dv^i}{dp_{J1}} \tag{12.6-69}$$

as the change in interface height with respect to the change in interface pressure. Then z_2^i is the linear function

$$z_2^i(J1) = z_0^i(J1) + \Delta p^i(J1)\frac{dz^i}{dp_{J1}} \tag{12.6-70}$$

Using Eq. (12.6-70) in Eq. (12.6-65), $v\frac{\partial \rho}{\partial z}$ can be expressed as a function of only Δp,

$$\begin{aligned} v\frac{\partial \rho}{\partial z} = \frac{1}{2}\left(v_1^i(J1) + \Delta v_0^i(J1) + \Delta p^i(J1)\frac{dv^i}{dp_{J1}}\right) \\ \times \frac{\rho_1(J1+1) + \Delta p(J1+1)\frac{d\rho}{dp_{J1+1}} - \rho_1^i(J1) - \Delta p^i\frac{d\rho^i}{dp_{J1}}}{z(J1+1) - z_0^i(J1) - \Delta p^i(J1)\frac{dz^i}{dp_{J1}}} \end{aligned} \tag{12.6-71}$$

12.6.2.2.4 Final Differenced Form of the Continuity Equation at the Lower Interface as a Function of Pressure and Mass Flow-Rate Changes

With $\frac{d\rho}{dt}$ in the interface continuity equation, Eq. (12.6-53), written in the form in Eq. (12.6-55) and $v\frac{\partial \rho}{\partial z}$ given by Eq. (12.6-71) and all other terms in Eq. (12.6-53) treated as in Section 12.6.2.1, the differenced form of the continuity equation at the lower interface is, after multiplication by $\left(z(J1+1) - z_2^i(J1)\right)$ and substitution of Eq. (12.6-70) for z_2^i,

$$\begin{aligned} &\Delta t\left(W_1(J1+1) - W_1^i(J1) + \Delta W(J1+1) - \Delta W^i(J1)\right) \\ &+\frac{A_c(J1)}{2}\left(\Delta\rho(J1+1)\frac{d\rho}{dp_{J1+1}} + \Delta\rho^i(J1)\frac{d\rho^i}{dp_{J1}}\right)\left(z(J1+1) - z_0^i(J1)\right. \\ &\left.-\Delta p^i(J1)\frac{dz^i}{dp_{J1}}\right) - \frac{A_c(J1)\Delta t}{2}\left(v_1^i(J1) + \Delta v_0^i(J1) + \Delta p^i(J1)\frac{dv^i}{dp_{J1}}\right) \\ &\times\left(\rho_1(J1+1) + \Delta p(J1+1)\frac{d\rho}{dp_{J1+1}} - \rho_1^i(J1) - \Delta p^i(J1)\frac{d\rho^i}{dp_{J1}}\right) \\ &= A_c(J1)\Delta t\frac{Q_0(J1)}{\lambda_1(J1)}\left[1 + \left(\Delta p^i(J1)\frac{dT^i}{dp_{J1}} + \Delta p(J1+1)\frac{dT}{dp_{J1+1}}\right)\frac{1}{2Q_0(J1)}\frac{dQ_0}{dT_{J1}}\right. \\ &\left.-\left(\Delta p^i(J1) + \Delta p(J1+1)\frac{1}{2\lambda_1(J1)}\frac{d\lambda}{dp_{J1}}\right)\right]\left(z(J1+1) - z_0^i(J1)\right. \\ &\left.+\Delta p^i(J1)\frac{dz^i}{dp_{J1}}\right) \end{aligned} \tag{12.6-72}$$

Setting $\Delta z_0(J1) = z(J1+1) - z_0^i(J1)$ and eliminating second-order terms gives a differenced interface continuity equation which is linear in the pressure and mass flow rate changes:

$$\begin{aligned}
&\Delta t\left(W_1(J+1) - W_1^i(J1) + \Delta W(J1+1) - \Delta W^i(J1)\right) \\
&+\frac{A_c(J1)}{2}\Delta z_0(J1)\left(\Delta p(J1+1)\frac{d\rho}{dp_{J1+1}} + \Delta p^i(J1)\frac{d\rho^i}{dp_{J1}}\right) \\
&-\frac{A_c(J1)\Delta t}{2}\Bigg\{\left[v_1^i(J1) + \Delta v_0^i(J1)\right]\left[\rho_1(J1+1) - \rho_1^i(J1)\right] \\
&+\left[v_1^i(J1) + \Delta v_0^i(J1)\right]\left[\Delta p(J1+1)\frac{d\rho}{dp_{J1+1}} - \Delta p^i(J1)\frac{d\rho^i}{dp_{J1}}\right] \\
&+\Delta p^i(J1)\frac{dv^i}{dp_{J1}}\left[\rho_1(J1+1) - \rho_1^i(J1)\right]\Bigg\} - A_c(J1)\Delta t\frac{Q_0(J1)}{\lambda_1(J1)}\Bigg\{\Delta z_0(J1) \\
&-\Delta p^i(J1)\frac{dz^i}{dp_{J1}} + \Delta z_0(J)\left[\Delta p^i(J1)\frac{dT^i}{dp_{J1}} + \Delta p(J1+1)\frac{dT}{dp_{J1+1}}\right]\frac{1}{2Q_0(J1)}\frac{dQ_0}{dT_{J1}} \\
&-\Delta z_0(J1)\frac{1}{2\lambda_1(J1)}\left[\Delta p^i(J1) + \Delta p(J1+1)\right]\frac{d\lambda}{dp_{J1}}\Bigg\} = 0
\end{aligned} \tag{12.6-73}$$

Eq. (12.6-73) can be simplified by defining the coefficients

$$\begin{aligned}
c_{1,J1} = \frac{A_c(J1)}{2}\Bigg\{&\Delta z_0(J1)\frac{d\rho^i}{dp_{J1}} + \Delta t\Bigg[\left(v_1^i(J1) + \Delta v_0^i(J1)\right)\frac{d\rho^i}{dp_{J1}} \\
&-\left(\rho_1(J1+1) - \rho_1^i(J1)\right)\frac{dv^i}{dp_J}\Bigg] + \Delta t\Bigg[\frac{2Q_0(J1)}{\lambda_1(J1)}\frac{dz^i}{dp_{J1}} \\
&-\frac{\Delta z_0(J1)}{\lambda_1(J1)}\frac{dQ_0}{dT_{J1}}\frac{dT^i}{dp_{J1}} + \frac{Q_0(J1)\Delta z_0(J1)}{\lambda_1^2(J1)}\frac{d\lambda}{dp_{J1}}\Bigg]\Bigg\}
\end{aligned} \tag{12.6-74}$$

$$c_{2,J1} = -\Delta t \tag{12.6-75}$$

$$\begin{aligned}
c_{3,J1} = \frac{A_c(J1)}{2}\Bigg\{&\Delta z_0(J1)\frac{d\rho}{dp_{J1+1}} - \Delta t\left(v_1^i(J1) + \Delta v_0^i(J1)\right)\frac{d\rho}{dp_{J1+1}} \\
&-\Delta t\frac{\Delta z_0(J1)}{\lambda_1(J1)}\left[\frac{dQ_0}{dT_{J1}}\frac{dT}{dp_{J1+1}} - \frac{Q_0(J1)}{\lambda_1(J1)}\frac{d\lambda}{dp_{J1}}\right]\Bigg\}
\end{aligned} \tag{12.6-76}$$

$$c_{4,J1} = \Delta t \tag{12.6-77}$$

and the source term

$$\begin{aligned}
h_{J1} = \Delta t\Bigg\{&W_1^i(J1) - W_1(J1+1) + A_c(J1)\Bigg[\frac{1}{2}\big(v_1^i(J1) \\
&+\Delta v_0^i(J1)\big)\left(\rho_1(J1+1) - \rho_1^i(J1)\right) + \frac{Q_0(J1)\Delta z_0(J1)}{\lambda_1(J1)}\Bigg]\Bigg\}
\end{aligned} \tag{12.6-78}$$

Then the differenced continuity equation at the lower interface can be written as

$$c_{1,J1}\Delta p^i(J1) + c_{2,J1}\Delta W^i(J1) + c_{3,J1}\Delta p(J1+1) + c_{4,J1}\Delta W(J1+1) = h_{J1} \tag{12.6-79}$$

Note that with v_1^i, Δv_0^i, $\frac{dv^i}{dp}$, and $\frac{dz^i}{dp}$ set to zero, the expressions for the coefficients $c_{1,J1}$, and the source term h_{J1} reduce to the forms given at the end of Section 12.6.2.1 for the continuity equation for a mesh segment which lies entirely within the bubble.

12.6.2.2.5 Differenced Form of the Continuity Equation at the Upper Interface

The derivation of the differenced continuity equation at the upper interface is nearly identical to that for the lower interface, and so just a brief explanation will be given here. The terms in the interface continuity equation of Eq. (12.6-53) are now expressed as follows:

The time derivative is

$$\begin{aligned}\frac{d\rho}{dt} &= \frac{\rho_2^i(J2)-\rho_1^i(J2)+\rho_2(J2-1)-\rho_1(J2-1)}{2\Delta t} \\ &= \frac{1}{2\Delta t}\left(\Delta p^i(J2)\frac{d\rho^i}{dp_{J2}}+\Delta p(J2-1)\frac{d\rho}{dp_{J2-1}}\right)\end{aligned} \tag{12.6-80}$$

The $v\frac{\partial\rho}{\partial z}$ term is

$$v\frac{\partial\rho}{\partial z} = \frac{v_1^i(J2)+\Delta v^i(J2)}{2}\frac{\rho_2^i(J2)-\rho_2(J2-1)}{z_2^i(J2)-z(J2-1)} \tag{12.6-81}$$

where

$$\Delta v^i(J2) = \frac{\Delta W_l\,(K+1)}{\rho_l^i(J2)A_c(J2-1)}f_{rwb}(K+1) \tag{12.6-82}$$

and

$$\Delta W_l\,(K+1) = \frac{\left(AA_0\,(K+1)+\theta_2\left(\Delta p^i(J2)-\Delta p^i\,(K+1)\right)\right)\Delta t}{I_1\,(K+1)+\theta_2\Delta I_1\,(K+1)+BB_0\,(K+1)\,\theta_2\Delta t} \tag{12.6-83}$$

The rewetting factor at the bottom of liquid slug $K+1$, $f_{rwb}\,(K+1)$, is computed as in Eq. (12.6-59). The expression for $\Delta v_i(J2)$ then reduces to

$$\Delta v^i(J2) = \Delta v_0^i(J2)+\Delta p^i(J2)\frac{dv^i}{dp_{J2}} \tag{12.6-84}$$

with the definitions of $\Delta v_0^i(J2)$ and $\frac{dv^i}{dp_{J2}}$ being set as were those for $\Delta v_0^i(J1)$ and $\frac{dv^i}{dp_{J1}}$. Also,

$$z_2^i(J2) = z_0^i(J2)+\Delta p^i(J2)\frac{dz^i}{dp_{J2}} \tag{12.6-85}$$

can be derived in exactly the same fashion as was Eq. (12.6-70). Therefore,

$$\begin{aligned}v\frac{\partial\rho}{\partial z} = \frac{1}{2}&\left(v_1^i(J2)+\Delta v_0^i(J2)+\Delta p^i(J2)\frac{dv^i}{dp_{J2}}\right) \\ \times&\left(\frac{\rho_1^i(J2)+\Delta p^i(J2)\frac{d\rho^i}{dp_{J2}}-\rho_1^i(J2-1)-\Delta p(J2-1)\frac{d\rho}{dp_{J2-1}}}{z_0^i(J2)+\Delta p^i(J2)\frac{dz^i}{dp_{J2}}-z(J2-1)}\right)\end{aligned} \tag{12.6-86}$$

Substituting Eq. (12.6-80) and Eq. (12.6-86) into Eq. (12.6-53) and expressing $\frac{\partial W}{\partial z}$ and $A_c\frac{Q}{\lambda}$ as in Section 12.6.2.1 gives the differenced upper interface continuity equation

$$\begin{aligned}c_{1,J2-1}\Delta p(J2-1)+c_{2,J2-1}\Delta W(J2-1)+c_{3,J2-1}\Delta p^i(J2)& \\ +c_{4,J2-1}\Delta W^i(J2) = h_{J2-1}&\end{aligned} \tag{12.6-87}$$

where

$$\begin{aligned}c_{1,J2-1} = \frac{A_c(J2-1)}{2}\Bigg\{&\Delta z_0(J2-1)\frac{d\rho}{dp_{J2-1}}+\Delta t\Bigg[\left(v_1^i(J2)+\Delta v_0^i(J2)\right)\frac{d\rho}{dp_{J2-1}} \\ &+\frac{\Delta z_0(J2-1)}{\lambda_1(J2-1)}\left(\frac{dQ_0}{dT_{J2-1}}\frac{dT}{dp_{J2-1}}-\frac{Q_0(J2-1)}{\lambda(J2-1)}\frac{d\lambda}{dp_{J2}}\right)\Bigg]\Bigg\}\end{aligned} \tag{12.6-88}$$

$$c_{2,J2-1} = -\Delta t \tag{12.6-89}$$

$$c_{3,J2-1} = \frac{A_c(J2-1)}{2}\Bigg\{\Delta z_0(J2-1)\frac{d\rho^i}{dp_{J2}} - \Delta t\Bigg[\left(v_1^i(J2)+\Delta v_0^i(J2)\right)\frac{d\rho^i}{dp_{J2}} + \left[\rho_1^i(J2)-\rho_1(J2-1)\right]\frac{dv^i}{dp_{J2}}\Bigg] - \Bigg[\Delta t\frac{2Q_0(J2-1)}{\lambda_1(J2-1)}\frac{dz^i}{dp_{J2}} + \frac{\Delta z_0(J2-1)}{\lambda_1(J2-1)}\left(\frac{dQ_0}{dT_{J2-1}}\frac{dT^i}{dp_{J2}} - \frac{Q_0(J2-1)}{\lambda_1(J2-1)}\frac{d\lambda}{dp_{J2}}\right)\Bigg]\Bigg\} \tag{12.6-90}$$

$$c_{4,J2-1} = \Delta t \tag{12.6-91}$$

$$h_{J2-1} = \Delta t\Bigg\{W_1(J2-1) - W_1^i(J2) + A_c(J2-1)\Bigg[\frac{1}{2}(v_1^i(J2) + \Delta v_0^i(J2))\left(\rho_1^i(J2)-\rho_1(J2-1)\right) + \frac{Q_0(J2-1)\Delta z_0(J2-1)}{\lambda_1(J2-1)}\Bigg]\Bigg\} \tag{12.6-92}$$

and

$$\Delta z_0(J2-1) = z_1^i(J2) - z(J2-1) \tag{12.6-93}$$

12.6.2.3 Finite Differencing of the Momentum Equation for a Mesh Segment Contained Entirely in One Bubble

Now that differenced forms of the continuity equation both within the bubble and at liquid-vapor interfaces have been derived, a similar process must be applied in the momentum equation. The differencing of the momentum equation is accomplished in essentially the same way as that of the continuity equation. Once again, bubble interface will be ignored for the sake of simplicity until the equations for an interior node have been developed.

12.6.2.3.1 Differencing of the Momentum Equation

The differenced forms of the terms in the momentum equation, Eq. (12.6-2), are a follows. The channel pressure drop is given by

$$\frac{\partial p}{\partial z}\Big|_{J+\frac{1}{2}} = \frac{p_2(J+1)-p_2(J)}{z(J+1)-z(J)} \tag{12.6-94}$$

The momentum convection takes the form

$$\frac{1}{A_c}\frac{\partial}{\partial z}\left(\frac{W^2}{\rho A_c}\right)\Bigg|_{J+\frac{1}{2}} = \frac{1}{A_c^2(J)\left(z(J+1)-z(J)\right)}\left(\frac{W_2^2(J+1)}{\rho_2(J+1)} - \frac{W_2^2(J)}{\rho_2(J)}\right) \tag{12.6-95}$$

The condensation term is expressed in terms of the contributions from cladding and structure as

$$F_c(J) = F_{ce}(J) + F_{cs}(J) \tag{12.6-96}$$

where the functions for F_{ce} and F_{cs} are linearly approximated as follows:

$$\text{if } Q_{e_2}(J) \geq 0,\ F_{ce}(J) = 0 \tag{12.6-97}$$

$$\text{if } Q_{e_2}(J) < 0,\ F_{ce}(J) = \frac{Q_{e_2}(J)}{2\lambda_2(J)A_c(J)}\left(\frac{W_2(J)}{\rho_2(J)} + \frac{W_2(J+1)}{\rho_2(J+1)}\right) \tag{12.6-98}$$

and

$$\text{if } Q_{s_2}(J) \geq 0,\ F_{cs}(J) = 0 \tag{12.6-99}$$

$$\text{if } Q_{s_2}(J) \leq 0,\ F_{cs}(J) = \frac{Q_{s_2}(J)}{2\lambda_2(J)A_c(J)}\left(\frac{W_2(J)}{\rho_2(J)} + \frac{W_2(J+1)}{\rho_2(J+1)}\right) \tag{12.6-100}$$

$Q_{e_2}(J)$ and $Q_{s_2}(J)$ are defined in Eq. (12.6-19). Now, linearize Q_{e_2} and Q_{s_2} by defining

$$Q_{oe}(J) = \gamma(J)\frac{T_{e_2}(J) - \overline{T}_1(1)}{R_{ec_2}(J)} \tag{12.6-101}$$

$$\frac{dQ_{oe}}{dT_J} = -\frac{\gamma(J)}{R_{ec_2}(J)} \tag{12.6-102}$$

$$Q_{os}(J) = \gamma(J)\gamma_2(J)\frac{T_{s_2}(J) - \overline{T}_1(J)}{R_{sc_2}(J)} \tag{12.6-103}$$

$$\frac{dQ_{os}}{dT_J} = -\frac{\gamma(J)\gamma_2(J)}{R_{sc_2}(J)} \tag{12.6-104}$$

These expressions are related to $Q_o(J)$, defined in Section 12.6.2.1.2, by

$$Q_o(J) = Q_{oe}(J) + Q_{os}(J) \tag{12.6-105}$$

and

$$\frac{dQ_o}{dT_J} = \frac{dQ_{oe}}{dT_J} + \frac{dQ_{os}}{dT_J} \tag{12.6-106}$$

Substituting these functions into the above formulations of F_{ce} and F_{cs} gives

$$\text{if } Q_{e_2}(J) \geq 0,\ F_{ce}(J) = 0 \tag{12.6-107}$$

$$\begin{aligned}\text{if } Q_{e_2}(J) < 0,\ F_{ce}(J) = {} & \frac{1}{2\lambda_2(J)A_c(J)}\left(\frac{W_2(J)}{\rho_2(J)} + \frac{W_2(J+1)}{\rho_2(J+1)}\right) \\ & \times \left[Q_{oe}(J) + \frac{1}{2}\frac{dQ_{oe}}{dT_J}\left(\Delta T(J) + \Delta T(J+1)\right)\right]\end{aligned} \tag{12.6-108}$$

$$\text{if } Q_{s_2}(J) \geq 0,\ F_{cs}(J) = 0 \tag{12.6-109}$$

$$\begin{aligned}\text{if } Q_{s_2}(J) < 0,\ F_{cs}(J) = {} & \frac{1}{2\lambda_2(J)A_c(J)}\left(\frac{W_2(J)}{\rho_2(J)} + \frac{W_2(J+1)}{\rho_2(J+1)}\right) \\ & \times \left[Q_{os}(J) + \frac{1}{2}\frac{dQ_{os}}{dT_J}\left(\Delta T(J) + \Delta T(J+1)\right)\right]\end{aligned} \tag{12.6-110}$$

If the definitions

$$F_{oe}(J) = \begin{cases} 0 & Q_{e_2} \geq 0 \\ Q_{oe}(J) & Q_{e_2}(J) < 0 \end{cases} \tag{12.6-111}$$

$$F_{os}(J) = \begin{cases} 0 & Q_{s_2} \geq 0, \\ Q_{os}(J) & Q_{s_2}(J) < 0 \end{cases} \tag{12.6-112}$$

$$F_o(J) = F_{oe}(J) + F_{os}(J) \tag{12.6-113}$$

are used in the expressions for F_{ce} and F_{cs}, the condensation term becomes

$$F_c(J) = \frac{F_o(J)}{2A_c(J)\lambda_2(J)}\left[1 + \left(\frac{F_{oc}(J)}{2Q_{oe}(J)F_o(J)}\frac{dQ_{oe}}{dT_J} + \frac{F_{os}(J)}{2Q_{os}(J)F_o(J)}\frac{dQ_{os}}{dT_J}\right) \times(\Delta t(J) + \Delta T(J+1))\right]\left(\frac{W_2(J)}{\rho_2(J)} + \frac{W_2(J+1)}{\rho_2(J+1)}\right) \tag{12.6-114}$$

This expression can be simplified by setting

$$\frac{dF_{oe}}{dT_J} = \begin{cases} \begin{cases} 0 & Q_{e_2}(J) \geq 0 \\ \frac{dQ_{oe}}{dT_J} & Q_{e_2}(J) < 0 \end{cases} \end{cases} \tag{12.6-115}$$

which can be stated more compactly as

$$\frac{dF_{oe}}{dT_J} = \frac{F_{oe}(J)}{Q_{oe}(J)}\frac{dQ_{oe}}{dT_J} \tag{12.6-116}$$

Similarly,

$$\frac{dF_{os}}{dT_J} = \frac{F_{os}(J)}{Q_{os}(J)}\frac{dQ_{os}}{dT_J} \tag{12.6-117}$$

so that the definition

$$\frac{dF_o}{dT_J} = \frac{dF_{oe}}{dT_J} + \frac{dF_{os}}{dT_J} \tag{12.6-118}$$

can be made. This gives

$$F_c(J) = \frac{F_o(J)}{2A_c(J)\lambda_2(J)}\left[1 + \frac{dF_o}{2F_o(J)}\frac{dF_o}{dT_J}(\Delta T(J) + \Delta T(J+1))\right] \times\left(\frac{W_2(J)}{\rho_2(J)} + \frac{W_2(J+1)}{\rho_2(J+1)}\right) \tag{12.6-119}$$

as the differenced form of the condensation term.

The change in mass flow rate with time is

$$\frac{\partial W}{\partial t} = \frac{W_2(J) + W_2(J+1) - W_1(J) - W_1(J+1)}{2\Delta t} \tag{12.6-120}$$

The orifice pressure drop becomes

$$\frac{W\,|W|}{2\rho_v A_c^2}\frac{\partial k_{or}}{\partial z} = \frac{1}{4A_c^2(J)}\left(\frac{W_2(J)\,|W_2(J)|}{\rho_2(J)} + \frac{W_2(J+1)\,|W_2(J+1)|}{\rho_2(J+1)}\right) \times\frac{k_{or}(J)}{z(J+1) - z(J)} \tag{12.6-121}$$

Finally, if the friction factor

$$f_v = A_{fr}\left(\frac{D_h W}{\mu A_c}\right)^{b_{fr}} \tag{12.6-122}$$

is considered in the differencing, the friction term may be written as

$$\frac{f_v W\,|W|}{2\rho A_c^2 D_h} = \frac{A_{fr} W\,|W|^{1+b_{fr}}}{2\rho\mu^{b_{fr}} D_h^{1-b_{fr}} A_c^{2+b_{fr}}}$$
$$= \frac{A_{fr}}{4D_h^{1-b_{fr}}(J)A_c^{2+b_{fr}}(J)}\left[\frac{W_2(J+1)\,|W_2(J+1)|^{1+b_{fr}}}{\rho_2(J+1)\mu_2^{b_{fr}}(J+1)} + \frac{W_2(J)\,|W_2(J)|^{1+b_{fr}}}{\rho_2(J)\mu_2^{b_{fr}}(J)}\right] \tag{12.6-123}$$

Using Eq. (12.6-94), Eq. (12.6-95), Eq. (12.6-119), Eq. (12.6-120), Eq. (12.6-121), and Eq. (12.6-123) results in a differenced momentum equation as follows:

$$\begin{aligned}
&\frac{p_2(J+1)-p_2(J)}{z(J+1)-z(J)}+\frac{1}{A_c^2(J)\left(z(J+1)-z(J)\right)}\left(\frac{W_2^2(J+1)}{\rho_2(J+1)}-\frac{W_2^2(J)}{\rho_2(J)}\right)\\
&+\frac{A_{fr}}{4D_h^{1-b_{fr}}(J)A_c^{2+b_{fr}}(J)}\left[\frac{W_2(J+1)\left|W_2(J+1)\right|^{1+b_{fr}}}{\rho_2(J+1)\mu_2^{b_{fr}}(J+1)}+\frac{W_2(J)\left|W_2(J)\right|^{1+b_{fr}}}{\rho_2(J)\mu_2^{b_{fr}}(J)}\right]\\
&-\frac{F_o(J)}{2A_c(J)\lambda_2(J)}\left[1+\frac{1}{2F_o(J)}\frac{dF_o}{dT_J}\left(\Delta T(J)+\Delta T(J+1)\right)\right]\left(\frac{W_2(J)}{\rho_2(J)}+\frac{W_2(J+1)}{\rho_2(J+1)}\right)\\
&+\frac{W_2(J)+W_2(J+1)-W_1(J)-W_1(J+1)}{2A_c(J)\Delta t}+\frac{1}{4A_c^2(J)}\Big(\frac{W_2(J)\left|W_2(J)\right|}{\rho_2(J)}\\
&+\frac{W_2(J+1)|W_2(J+1)|}{\rho_2(J+1)}\Big)\frac{k_{or}(J)}{z(J+1)-z(J)}=0
\end{aligned}\tag{12.6-124}$$

The equation must now be linearized and reduced to a linear function of Δp and ΔW; this procedure will be discussed in the next subsection.

12.6.2.3.2 Linearization of Advanced Time Quantities and Introduction of Saturation Conditions

Eq. (12.6-124) is made linear by expressing the advanced time variables in linearized form as in Eq. (12.6-21). This means that, in addition to using the expressions in Eq. (12.6-22) through Eq. (12.6-25) for density, heat of vaporization, mass flow rate, and temperature, the linearized form of Eq. (12.6-124) must include the following representations of pressure and viscosity:

$$p_2(J)=p_1(J)+\Delta p(J)\tag{12.6-125}$$

$$\mu_2(J)=\mu_1(J)+\Delta\mu(J)\tag{12.6-126}$$

The differenced momentum equation therefore becomes

$$\begin{aligned}
&\frac{p_1(J+1)+\Delta p(J+1)-p_1(J)-\Delta p(J)}{z(J+1)-z(J)}+\frac{1}{A_c^2(J)\left(z(J+1)-z(J)\right)}\\
&\times\left(\frac{\left(W_1(J+1)+\Delta W(J+1)\right)^2}{\rho_1(J+1)+\Delta\rho(J+1)}-\frac{\left(W_1(J)+\Delta W(J)\right)^2}{\rho_1(J)+\Delta\rho(J)}\right)\\
&+\frac{A_{fr}}{4D_h^{1-b_{fr}}(J)A_c^{2+b_{fr}}(J)}\left(\frac{\left(W_1(J+1)+\Delta W(J+1)\right)\left|W_1(J+1)+\Delta W(J+1)\right|^{1+b_{fr}}}{\left(\rho_1(J+1)+\Delta\rho(J+1)\right)\left(\mu_1(J+1)+\Delta\mu(J+1)\right)^{b_{fr}}}\right.\\
&\left.+\frac{\left(W_1(J)+\Delta W(J)\right)\left|W_1(J)+\Delta W(J)\right|^{1+b_{fr}}}{\left(\rho_1(J)+\Delta\rho(J)\left(\mu_1(J)+\Delta\mu(J)\right)^{b_{fr}}\right.}\right)-\frac{F_o(J)}{2A_c(J)(\lambda_1(J)+\Delta\lambda(J))}\\
&\times\left(1+\frac{1}{2F_o(J)}\frac{dF_o}{dT_J}\left(\Delta T(J)+\Delta T(J+1)\right)\right)\left(\frac{W_1(J)+\Delta W(J)}{\rho_1(J)+\Delta\rho(J)}+\frac{W_1(J+1)+\Delta W(J+1)}{\rho_1(J+1)+\Delta\rho(J+1)}\right)\\
&+\frac{\Delta W(J)+\Delta W(J+1)}{2A_c(J)\Delta t}+\frac{1}{4A_c^2(J)}\left(\frac{\left(W_1+\Delta W(J)\right)\left|W_1(J)+\Delta W(J)\right|}{\rho_1(J)+\Delta\rho(J)}\right.\\
&\left.+\frac{\left(W_1(J+1)+\Delta W(J+1)\right)\left|W_1(J+1)+\Delta W(J+1)\right|}{\rho_1(J+1)+\Delta\rho(J+1)}\right)\frac{k_{or}(J)}{z(J+1)-z(J)}=0
\end{aligned}\tag{12.6-127}$$

Note that the orifice coefficient k_{or} and the hydraulic diameter D_h are, like the channel area A_c, considered to be constant over the time step. Now, Eq. (12.6-127) can be expressed in terms of only two independent variables, Δp and ΔW, by imposing the restriction that ρ, T, λ, and μ be functions of p only, i.e., by imposing saturation conditions. This is accomplished by substituting Eq. (12.6-36) through Eq. (12.6-38), as well as the expression

$$\Delta\mu(J) = \Delta p(J)\frac{d\mu}{dp_J} \tag{12.6-128}$$

into Eq. (12.6-127). The resulting equation, after multiplying by $\Delta t\,(z(J+1) - z(J))$ and letting $\Delta z(J) = z(J+1) - z(J)$, is the very long expression

$$\begin{aligned}
&(p_1(J+1) - p_1(J))\,\Delta t + \Delta t\Delta p(J+1) - \Delta t\Delta p(J) \\
&+\frac{\Delta t}{A_c^2(J)}\left[\frac{W_1^2(J+1) + 2W_1(J+1)\Delta W(J+1) + (\Delta W(J+1))^2}{\rho_1(J+1)}\left(1 - \frac{\Delta p(J+1)}{\rho_1(J+1)}\frac{d\rho}{dp_{J+1}}\right)\right. \\
&\left.-\frac{W_1^2(J) + 2W_1(J)\Delta W(J) + (\Delta W(J))^2}{\rho_1(J)}\left(1 - \frac{\Delta p(J)}{\rho_1(J)}\frac{d\rho}{dp_J}\right)\right] \\
&+\frac{A_{fr}\Delta t\Delta z(J)}{4D_h^{1-b_{fr}}(J)A_c^{2+b_{fr}}(J)}\left[\frac{W_1(J+1)\,|W_1(J+1)|^{1+b_{fr}}\left(\frac{1+(2+b_{fr})\Delta W(J+1)}{W_1(J+1)}\right)}{\rho_1(J+1)\mu_1^{b_{fr}}(J+1)}\right. \\
&\times\left(1 - \frac{\Delta p(J+1)}{\rho_1(J+1)}\frac{d\rho}{dp_{J+1}} - b_{fr}\frac{\Delta p(J+1)}{\mu_1(J+1)}\frac{d\mu}{dp_{J+1}}\right) \\
&\left.+\frac{W_1(J)\,|W_1(J)|^{1+b_{fr}}\left(1 + \frac{(2+b_{fr})\Delta W(J)}{W_1(J)}\right)}{\rho_1(J)\mu_1^{b_{fr}}(J)}\left(1 - \frac{\Delta p(J)}{\rho_1(J)}\frac{d\rho}{dp_J} - b_{fr}\frac{\Delta p(J)}{\mu_1(J)}\frac{d\mu}{dp_J}\right)\right] \\
&-\frac{F_o(J)\Delta t\Delta z(J)}{2A_c(J)\lambda_1(J)}\left(1 - \frac{\Delta p(J) + \Delta p(J+1)}{2\lambda_1(J)}\frac{d\lambda}{dp_J}\right) \\
&\times\left[1 + \frac{1}{2F_o(J)}\frac{dF_o}{dT_J}\left(\Delta p(J)\frac{dT}{dp_J} + \Delta p(J+1)\frac{dT}{dp_{J+1}}\right)\right] \\
&\times\left[\frac{W_1(J+1)}{\rho_1(J)}\left(1 + \frac{\Delta W(J)}{W_1(J)}\right)\left(1 - \frac{\Delta p(J)}{\rho_1(J)}\frac{d\rho}{dp_J}\right)\right. \\
&\left.+\frac{W_1(J+1)}{\rho_1(J+1)}\left(1 + \frac{\Delta W(J+1)}{W_1(J+1)}\right)\left(1 - \frac{\Delta p(J+1)}{\rho_1(J+1)}\frac{d\rho}{dp_{J+1}}\right)\right] \\
&+\frac{\Delta z(J)}{2A_c(J)}\Delta W(J) + \frac{\Delta z(J)}{2A_c(J)}\Delta W(J+1) \\
&+\frac{\Delta t}{4A_c^2(J)}\left[\frac{W_1(J)\,|W_1(J)|\left(1 + \frac{2\Delta W(J)}{W_1(J)}\right)}{\rho_1(J)}\left(1 - \frac{\Delta p(J)}{\rho_1(J)}\frac{d\rho}{dp_J}\right)\right. \\
&\left.+\frac{W_1(J+1)\,|W_1(J+1)|\left(1 + \frac{2\Delta W(J+1)}{W_1(J+1)}\right)}{\rho_1(J+1)}\left(1 - \frac{\Delta p(J+1)}{\rho_1(J+1)}\frac{d\rho}{dp_{J+1}}\right)\right] \\
&\times k_{or}(J) = 0
\end{aligned} \tag{12.6-129}$$

where it is assumed that $\Delta\mu \ll \mu_1$ and $\Delta\rho \ll \rho_1$, so that

$$\frac{1}{\rho_1(J) + \Delta\rho(J)} = \frac{1}{\rho_1(J)}\left(1 - \frac{\Delta\rho(J)}{\rho_1(J)}\right) \tag{12.6-130}$$

and

$$\frac{1}{\left(\mu_1(J)+\Delta\mu(J)\right)^{b_{fr}}}=\frac{1}{\mu_1^{b_{fr}}(J)}\left(1-b_{fr}\frac{\Delta\mu(J)}{\mu_1(J)}\right) \tag{12.6-131}$$

Eliminating second-order terms produces the final linearized form of the differenced momentum equation:

$$b_{1,J}\Delta p(J)+b_{2,J}\Delta W(J)+b_{3,J}\Delta p(J+1)+b_{4,J}\Delta W(J+1)=g_J \tag{12.6-132}$$

with

$$\begin{aligned}b_{1,J}=-\Delta t+\frac{\Delta t}{A_c^2(J)}\frac{W_1^2(J)}{\rho_1^2(J)}\frac{d\rho}{dp_J}-\frac{A_{fr}\Delta t\Delta z(J)}{4D_h^{1-b_{fr}}(J)A_c^{2+b_{fr}}(J)}\frac{W_1(J)\left|W_1(J)\right|^{1+b_{fr}}}{\rho_1(J)\mu_1^{b_{fr}}(J)}\\ \times\left(\frac{1}{\rho_1(J)}\frac{d\rho}{dp_J}+\frac{b_{fr}}{\mu_1(J)}\frac{d\mu}{dp_J}\right)\\ +\frac{F_o(J)\Delta t\Delta z(J)}{4A_c(J)\lambda_1(J)}\left[\frac{1}{\lambda_1(J)}\frac{d\lambda}{dp_J}\left(\frac{W_1(J)}{\rho_1(J)}+\frac{W_1(J+1)}{\rho_1(J+1)}\right)\right.\\ \left.-\frac{1}{F_o(J)}\frac{dF_o}{dT_J}\frac{dT}{dp_J}\left(\frac{W_1(J)}{\rho_1(J)}+\frac{W_1(J+1)}{\rho_1(J+1)}\right)+\frac{2W_1(J)}{\rho_1^2(J)}\frac{d\rho}{dp_J}\right]\\ -\frac{W_1(J)\left|W_1(J)\right|\Delta t}{4A_c^2(J)\rho_1^2(J)}\frac{d\rho}{dp_J}k_{or}(J)\end{aligned} \tag{12.6-133}$$

$$\begin{aligned}b_{2,J}=-\frac{2W_1(J)\Delta t}{\rho_1(J)A_c^2(J)}+\frac{A_{fr}\Delta t\Delta z(J)}{4D_h^{1-b_{fr}}(J)A_c^{2+b_{fr}}(J)}\frac{(2+b_{fr})\left|W_1(J)\right|^{1+b_{fr}}}{\rho_1(J)\mu_1^{b_{fr}}(J)}\\ -\frac{F_o(J)\Delta t\Delta z(J)}{2A_c(J)\lambda_1(J)\rho_1(J)}+\frac{\Delta z(J)}{2A_c(J)}+\frac{\Delta t\left|W_1(J)\right|}{2A_c^2(J)\rho_1(J)}k_{or}(J)\end{aligned} \tag{12.6-134}$$

$$\begin{aligned}b_{3,J}=\Delta t-\frac{\Delta t}{A_c^2(J)}\frac{W_1^2(J+1)}{\rho_1^2(J+1)}\frac{d\rho}{dp_{J+1}}-\frac{A_{fr}\Delta t\Delta z(J)}{4D_h^{1-b_{fr}}(J)A_c^{2+b_{fr}}(J)}\frac{W_1(J+1)\left|W_1(J+1)\right|^{1+b_{fr}}}{\rho_1(J+1)\mu_1^{b_{fr}}(J+1)}\\ \times\left(\frac{1}{\rho_1(J+1)}\frac{d\rho}{dp_{J+1}}+\frac{b_{fr}}{\mu_1(J+1)}\right)\frac{d\mu}{dp_{J+1}}+\frac{F_o(J)\Delta t\Delta z(J)}{4A_c(J)\lambda_1(J)}\\ \times\left[\frac{1}{\lambda_1(J)}\frac{d\lambda}{dp_J}\left(\frac{W_1(J)}{\rho_1(J)}+\frac{W_1(J+1)}{\rho_1(J+1)}\right)\right.\\ \left.-\frac{1}{F_o(J)}\frac{dF_o}{dT_J}\frac{dT}{dp_{J+1}}\left(\frac{W_1(J)}{\rho_1(J)}+\frac{W_1(J+1)}{\rho_1(J+1)}\right)-\frac{2W_1(J+1)}{\rho_1^2(J+1)}\frac{d\rho}{dp_{J+1}}\right]\\ -\frac{W_1(J+1)\left|W_1(J+1)\right|\Delta t}{4A_c^2(J)\rho_1^2(J+1)}\frac{d\rho}{dp_{J+1}}k_{or}(J)\end{aligned} \tag{12.6-135}$$

$$\begin{aligned}b_{4,J}=-\frac{2W_1(J+1)\Delta t}{\rho_1(J+1)A_c^2(J)}+\frac{A_{fr}\Delta t\Delta z(J)}{4D_h^{1-b_{fr}}(J)A_c^{2+b_{fr}}(J)}\frac{(2+b_{fr})\left|W_1(J+1)\right|^{1+b_{fr}}}{\rho_1(J+1)\mu_1^{b_{fr}}(J+1)}\\ -\frac{F_o(J)\Delta t\Delta z(J)}{2A_c(J)\lambda_1(J)\rho_1(J+1)}+\frac{\Delta z(J)}{2A_c(J)}+\frac{\Delta t\left|W_1(J+1)\right|}{2A_c^2(J)\rho_1(J+1)}k_{or}(J)\end{aligned} \tag{12.6-136}$$

$$
\begin{aligned}
g_J = (p_1(J) - p_1(J+1))\,\Delta t - \frac{\Delta t}{A_c^2(J)}\left(\frac{W_1^2(J+1)}{\rho_1(J+1)} - \frac{W_1^2(J)}{\rho_1(J)}\right) \\
- \frac{A_{fr}\Delta t \Delta z(J)}{4D_h^{1-b_{fr}}(J)A_c^{2+b_{fr}}(J)}\left(\frac{W_1(J+1)\,|W_1(J+1)|^{1+b_{fr}}}{\rho_1(J+1)\mu_1^{b_{fr}}(J+1)} + \frac{W_1(J)\,|W_1(J)|^{1+b_{fr}}}{\rho_1(J1)\mu_1^{b_{fr}}(J1)}\right) \\
+ \frac{F_o(J)\Delta t \Delta z(J)}{2A_c(J)\lambda_1(J)}\left(\frac{W_1(J)}{\rho_1(J)} + \frac{W_1(J+1)}{\rho_1(J+1)}\right) \\
- \frac{\Delta t k_{or}(J)}{4A_c^2(J)}\left(\frac{W_1(J)\,|W_1(J)|}{\rho_1(J)} + \frac{W_1(J+1)\,|W_1(J+1)|}{\rho_1(J+1)}\right)
\end{aligned}
\tag{12.6-137}
$$

12.6.2.4 Finite Differencing of the Momentum Equation for a Mesh Segment Which Contains a Bubble Interface

The inclusion of a bubble interface in node J introduces the Lagrangian total derivative, as described in Section 12.6.2.2.1. The momentum equation then becomes

$$
\frac{\partial p}{\partial z} + \frac{1}{A_c^2}\frac{\partial}{\partial z}\frac{W^2}{\rho_v} + \frac{f_v W|W|}{2\rho_v D_h A_c^2} - F_c + \frac{1}{A_c}\frac{dW}{dt} - v\frac{\partial W}{\partial z} + \frac{W\,|W|}{2\rho_v A_c^2}\frac{\partial k_{or}}{\partial z} = 0 \tag{12.6-138}
$$

As in the case of the continuity equation, the impact of the bubble interface on the differenced equation is transmitted via the interface velocity v^i and the interface position z^i. Therefore, only those terms in Eq. (12.6-138) which include either v^i or z^i will alter the coefficient b_i or the source term g in Eq. (12.6-132).

12.6.2.4.1 Differenced Form of the Momentum equation at the Lower Interface

The terms of Eq. (12.6-138) can be differenced as in Section 12.6.2.3; in addition, the term $v\frac{\partial W}{\partial z}$ can be expressed as

$$
v\frac{\partial W}{\partial z} = \frac{v_2^i(J1)}{2}\frac{W_2(J1+1) - W_2^i(J1)}{\Delta z^i(J1)} \tag{12.6-139}
$$

where $\Delta z^i(J1) = z(J1+1) - z_2^i(J)$. Therefore, the differenced form of Eq. (12.6-138) is

$$
\begin{aligned}
\frac{p_2(J1+1) - p_2^i(J1)}{\Delta z^i(J1)} + \frac{1}{A_c^2(J1)\Delta z^i(J1)}\left(\frac{W_2^2(J1+1)}{\rho_2(J1+1)} - \frac{W_2^2(J1)}{\rho_2(J1)}\right) \\
+ \frac{A_{fr}}{4D_h^{1-b_{fr}}(J1)A_c^{2+b_{fr}}(J1)}\left(\frac{W_2(J1+1)\,|W_2(J1+1)|^{1+b_{fr}}}{\rho_2(J1+1)\mu_2^{b_{fr}}(J1+1)} + \frac{W_2^i(J1)\,\left|W_2^i(J1)^{1+b_{fr}}\right|}{\rho_2(J1)\mu_2^{b_{fr}}(J1)}\right) - F_e(J1) \\
+ \frac{W_2(J1+1) + W_2^i(J1) - W_1(J1+1) - W_1^i(J1)}{2A_c(J1)\Delta t} - \frac{v_2^i(J1)}{2A_c(J1)}\frac{W_2^i(J1+1) - W_2^i(J1)}{\Delta z^i(J1)} \\
+ \frac{1}{4A_c^2(J1)}\left(\frac{W_2^i(J1)\,\left|W_2^i(J1)\right|}{\rho_2(J1)} + \frac{W_2(J1+1)\,|W_2(J1+1)|}{\rho_2(J1+1)}\right)\frac{k_{or}^i(J1)}{\Delta z(J1)} = 0
\end{aligned}
\tag{12.6-140}
$$

Multiplying Eq. (12.6-140) by $\Delta t \Delta z^i(J1)$ eliminates $z_2^i(J1)$ from all but the third, fourth, and fifth terms. The sixth term is the only one which includes $v_2^i(J1)$. Thus, only these four terms need to be examined to determine the contribution of the bubble interface to the final differenced momentum equation.

12.6.2.4.2 Specification of Bubble Interface Terms

Of the four terms, all but the one involving $v_2^i(J1)$ are present in the momentum equation as derived in Section 12.6.2.3. They affect the interface contribution only because they involve the factor $z\Delta^i(J1)$. For example, consider the condensation term:

$$\Delta t \Delta z^i(J1) F_c(J1) = \Delta t F_c(J1) \Delta z_0^i(J1) - \Delta p^i(J1) \frac{dz^i}{dp_{J1}} \Delta t F_c(J1) \tag{12.6-141}$$

The first term on the right-hand side is identical to the condensation term as expressed in the differenced momentum equation, Eq. (12.6-132). Therefore, the condensation term contribution to the momentum equation is modified to account for the bubble interface simply by adding the second term on the right-hand side of Eq. (12.6-140) to Eq. (12.6-132). Written out in full, this term is, after linearization of advanced time quantities and substitution of the saturation condition relations,

$$\begin{aligned}
\Delta p^i(J1) \frac{dz^i}{dp_{J1}} \Delta t F_c(J1) = \Delta p^i(J1) \frac{dz^i}{dp_{J1}} \frac{F_o(J1)\Delta t}{2A_c(J1)\lambda_1(J1)} \\
\times \left(1 - \frac{\Delta p^i(J1) + \Delta p(J1+1)}{2\lambda_i(J1)} \frac{d\lambda}{dp_{J1}}\right) \\
\times \left(1 + \frac{2}{F_o(J1)} \frac{dF_o}{dT_{J1}} \left(\Delta p(J1) \frac{dT^i}{dp_{J1}} + \Delta p(J1+1) \frac{dT}{dp_{J1+1}}\right)\right) \\
\times \left(\frac{W_1(J1)}{\rho_1(J1)} \left(1 + \frac{\Delta W^i(J1)}{W_1^i(J1)}\right) \left(1 - \frac{\Delta p^i(J1)}{\rho_1(J1+1)} \frac{d\rho}{dp_{J1+1}}\right)\right. \\
\left. + \frac{W_1(J1+1)}{\rho_1(J1+1)} \left(1 + \frac{\Delta W(J1+1)}{W_1(J1+1)}\right) \left(1 - \frac{\Delta p(J1+1)}{\rho_1(J1+1)} \frac{d\rho}{dp_{J1+1}}\right)\right)
\end{aligned} \tag{12.6-142}$$

Neglecting second-order terms, this reduces to

$$\Delta p^i(J1) \frac{dz^i}{dp_{J1}} \Delta t F_c(J1) = \Delta p^i(J1) \frac{dz^i}{dp_{J1}} \frac{F_o(J1)\Delta t}{2A_c(J1)\lambda_1(J1)} \left(\frac{W_1^i(J1)}{\rho_1(J1)} + \frac{W_1(J1+1)}{\rho_1(J1+1)}\right) \tag{12.6-143}$$

Similarly, the friction term contribution comes from

$$\begin{aligned}
\Delta p^i(J1) \frac{dz^i}{dp_{J1}} \Delta t \frac{A_{fr}}{4D_h^{1-b_{fr}}(J1) A_c^{2+b_{fr}}(J1)} \left(\frac{W_2(J1+1)\left|W_2(J1+1)\right|^{1+b_{fr}}}{\rho_2(J1+1)\mu_2^{b_{fr}}(J1+1)}\right. \\
\left. + \frac{W_2^i(J1)\left|W_2^i(J1)\right|^{1+b_{fr}}}{\rho_1(J1)\mu_2^{b_{fr}(J1)}}\right) = \Delta p^i(J1) \frac{dz^i}{dp_{J1}} \frac{A_{fr}\Delta t}{4D_h^{1-b_{fr}}(J1) A_c^{2+b_{fr}}(J1)} \\
\times \left(\frac{W_1(J1+1)\left|W_1(J1+1)\right|^{1+b_{fr}}}{\rho_1(J1+1)\mu_2^{b_{fr}}(J1+1)} + \frac{W_1^i(J1)\left|W_1^i(J1)\right|^{1+b_{fr}}}{\rho_1(J1)\mu_1^{b_{fr}}(J1)}\right)
\end{aligned} \tag{12.6-144}$$

if second-order terms are dropped. The time derivative of the mass flow rate is

$$\begin{aligned}
\Delta t \Delta z^i(J1) \frac{\Delta W(J1+1) - \Delta W^i(J1)}{2A_c(J1)\Delta t} \\
= \left(\Delta z_o^i(J1) - \Delta p^i(J1) \frac{dz^i}{dp_{J1}}\right) \frac{\Delta W(J1+1) + \Delta W^i(J1)}{2A_c(J1)} \\
= \frac{\Delta z_o^i(J1)}{2A_c(J1)} \left(\Delta W(J1+1) + \Delta W^i(J1)\right)
\end{aligned} \tag{12.6-145}$$

if second-order terms are dropped. This is identical to the differenced form of $\frac{1}{A_c}\frac{\partial W}{\partial t}$ for mesh segments which are contained within the bubble and so this term actually makes no contribution with respect to the bubble interface.

The last term to analyze is $v\frac{\partial W}{\partial z}$. This reduces to

$$\begin{aligned}
&\frac{v_2^i(J1)\Delta t}{2A_c(J1)}\left(W_2^i(J1+1)-W_2^i(J1)\right)\\
&=\frac{\Delta t}{2A_c(J1)}\left(v_1^i(J1)+\Delta v_0^i(J1)+\Delta p^i(J1)\frac{dv_i}{dp_{J1}}\right)\\
&\times\left(W_1(J1+1)+\Delta W(J1+1)-W_1^i(J1)-\Delta W^i(J1)\right)\\
&=\frac{\Delta t}{2A_c(J1)}\left(v_1^i(J1)+\Delta v_0^i(J1)\right)\left(W_1(J1+1)-W_1^i(J1)\right)\\
&+(v_1^i(J1)+\Delta v_0^i(J1))\Delta W(J1+1)-(v_1^i(J1)+\Delta v_0^i(J1))\Delta W^i(J1)\\
&+\Delta p^i(J1)\frac{dv^i}{dp_{J1}}(W_1(J1+1)-W_1^i(J1))
\end{aligned}\tag{12.6-146}$$

Adding Eq. (12.6-143), Eq. (12.6-144) and Eq. (12.6-146) to Eq. (12.6-132) produces the final differenced momentum equation at the lower interface:

$$\widetilde{b}_{1,J1}\Delta p^i(J1)+\widetilde{b}_{2,J1}\Delta W^i(J1)+\widetilde{b}_{3,J1}\Delta p(J1+1)+\widetilde{b}_{4,J1}\Delta W(J1+1)=\widetilde{g}_{J1}\tag{12.6-147}$$

where

$$\begin{aligned}
\widetilde{b}_{1,J1}=b_{1,J1}&+\frac{dz^i}{dp_{J1}}\left[\frac{F_o(J1)\Delta t}{2A_c(J1)\lambda_1(J1)}\left(\frac{W_1^i(J1)}{\rho_1(J1)}+\frac{W_1(J1+1)}{\rho_1(J1+1)}\right)\right.\\
&-\frac{A_{fr}\Delta t}{4D_h^{1-b_{fr}}(J1)A_c^{2+b_{fr}}(J1)}\left(\frac{W_1(J1+1)\left|W_1(J1+1)\right|^{1+b_{fr}}}{\rho_1(J1+1)\mu_1^{b_{fr}}(J1+1)}\right.\\
&\left.\left.+\frac{W_1^i(J1)\left|W_1^i(J1)\right|^{1+b_{fr}}}{\rho_1(J1)\mu_1^{b_{fr}}(J1)}\right)\right]\\
&-\frac{dv^i}{dp_{J1}}\frac{\Delta t}{2A_c(J1)}\left(W_1(J1+1)-W_1^i(J1)\right)
\end{aligned}\tag{12.6-148}$$

$$\widetilde{b}_{2,J1}=b_{2,J1}+\frac{\Delta t}{2A_c(J1)}\left(v_1^i(J1)+\Delta v_0^i(J1)\right)\tag{12.6-149}$$

$$\widetilde{b}_{3,J1}=b_{3,J1}\tag{12.6-150}$$

$$\widetilde{b}_{4,J1}=b_{4,J1}-\frac{\Delta t}{2A_c(J1)}\left(v_1^i(J1)+\Delta v_0^i(J1)\right)\tag{12.6-151}$$

$$\widetilde{g}_{J1}=g_{J1}+\frac{\Delta t}{2A_c(J1)}\left(v_1^i(J1)+\Delta v_0^i(J1)\right)\left(W_1(J1+1)-W_1^i(J1)\right)\tag{12.6-152}$$

with $b_{1,J1}$, $b_{2,J1}$, $b_{3,J1}$, $b_{4,J1}$, and g_{J1} being defined as in Eq. (12.6-132).

12.6.2.4.3 Differenced Form of the Momentum Equation at the Upper Interface

The development of the momentum equation at the upper interface is identical to that explained in Section 12.6.2.4.1 and Section 12.6.2.4.2 for the lower interface, and so only the final equation will be given here:

$$\widetilde{b}_{1,J2-1}\Delta p(J2-1)+\widetilde{b}_{2,J2-1}\Delta W(J2-1)+\widetilde{b}_{3,J2-1}\Delta p^i(J2) + \widetilde{b}_{4,J2-1}\Delta W^i(J2)=\widetilde{g}_{J2-1} \tag{12.6-153}$$

with

$$\widetilde{b}_{1,J2-1}=b_{1,J2-1} \tag{12.6-154}$$

$$\widetilde{b}_{2,J2-1}=b_{2,J2-1}+\frac{\Delta t}{2A_c(J2-1)}\left(v_1^i(J2)+\Delta v_0^i(J2)\right) \tag{12.6-155}$$

$$\begin{aligned}\widetilde{b}_{3,J2-1}=&\,b_{3,J2-1}-\frac{\Delta t}{2A_c(J2-1)}\frac{dv^i}{dp_{J2}}\left(W_1^i(J2)-W_1(J2-1)\right)\\ &+\frac{A_{fr}\Delta t}{4D_h^{1-b_{fr}}(J2-1)A_c^{2+b_{fr}}(J2-1)}\frac{dz^i}{dp_{J2}}\left(\frac{W_1^i(J2)\left|W_1^i(J2)\right|^{1+b_{fr}}}{\rho_1(J2)\mu_1^{b_{fr}}(J2)}\right.\\ &\left.+\frac{W_1(J2-1)\left|W_1(J2-1)\right|^{1+b_{fr}}}{\rho_1(J2)\mu_1^{b_{fr}}(J2)}\right)\\ &-\frac{\Delta t}{2A_c(J2-1)}\frac{dz^i}{dp_{J2}}\frac{F_o(J2-1)}{\lambda_1(J2-1)}\left(\frac{W_1^i(J2)}{\rho_1(J2)}+\frac{W_1(J2-1)}{\rho_1(J2-1)}\right)\end{aligned} \tag{12.6-156}$$

$$\widetilde{b}_{4,J2-1}=b_{4,J2-1}-\frac{\Delta t}{2A_c(J2-1)}v_1^i(J2)+\Delta v_0^i(J2) \tag{12.6-157}$$

$$\widetilde{g}_{J2-1}=g_{J2-1}+\frac{\Delta t}{2A_c(J2-1)}\left(v_1^i(J2)+\Delta v_0^i(J2)\right)\left(W_1^i(J2)-W_1^i(J-2)\right) \tag{12.6-158}$$

12.6.2.5 Additional Details Concerning Interface Nodes

Because at least one of the boundaries of an interface mesh segment is not usually aligned with the fixed segment boundaries of the axial mesh, variables defined at the interface segment midpoint must be handled somewhat differently from what they are for a segment located in the interior of a bubble. Midpoint quantities such as A_c and D_h, which are considered to be piecewise constant over the fixed mesh segments, are assigned the values of the fixes mesh segment in which the bubble interface lies. Thus, for example, the flow area at the upper interface is taken as $A_c(J2-1)$ because the upper interface lies in axial mesh segment $J2-1$. This convention has been used throughout the equations in Section 12.6.2.2 and Section 12.6.2.4. However, three midpoint variables are exceptions to this rule: the cladding temperature T_e, the structure temperature T_s, and the average coolant temperature $\overline{T}$. T_e and T_s are calculated outside the two-phase flow model once every heat-transfer time step. As used in the boiling calculation, both temperatures are extrapolated values calculated using the temperatures at the beginning and the end of the last heat-transfer time step and extrapolated to the end of the current coolant time step. All three temperatures are taken to be located at the midpoints of the Eulerian grid intervals, whereas the boiling calculation needs them at the midpoints of the interface vapor mesh segments (for example, T_e, T_s, and $\overline{T}$ are needed at a point halfway between $z(J2-1)$ and the upper interface). Therefore, T_e and T_s are interpolated in space to the center of the interface segments using the values in the interface fixed mesh segments and the adjacent vapor segments, while $\overline{T}$ is interpolated to the points using the temperatures at the interfaces and at the nearest fixed mesh boundaries contained in the bubble.

Special considerations must be given to any mesh boundary that is crossed by a liquid-vapor interface during a time step. When setting up the interior segments for the pressure-drop calculation, any segment which is outside

the bubble at either the beginning or the end of the time step is bypassed and treated as part of the interface segment. Also, if at the end of the step the interface is within ε of a segment boundary ($\varepsilon = .002$) then the segment is also bypassed. These tests are all made on the basis of $z_1^i(J)$ and $z_0^i(J)$ as given by Eq. (12.6-68). Therefore, a segment JC is included as an interior segment only if

$$z_1^i(J1) < z(JC)$$

$$z_0^i(J1) < z\left(JC\right) - \varepsilon$$

$$z_1^i(J2) > z(JC+1)$$

and

$$z_0^i(J2) > z\left(JC+1\right) + \varepsilon$$

After the calculations of Δp and ΔW for segments inside the bubble are completed, p_2 and W_2 for bypassed segments are obtained by linear interpolation between the interface and the first segment inside the interface.

If an interface segment, $JT = J1$ or $JT = J2$, represents parts of more than one fixed Eulerian mesh segment, then $A_c(JT)$ and $D_h(JT)$ for the interface segment are obtained as weighted averages over the Eulerian segment values. The weighting factors are proportional to the Δz within the bubble. Also, $1/D_h$ is averaged:

$$\frac{1}{D_h(JT)} = \frac{\sum_{\text{JC}} \frac{\Delta z(JC)}{D_h(JC)}}{\sum_{\text{JC}} \Delta z(JC)} \tag{12.6-159}$$

The $\Delta z(JC)$s are based on $z_0^i(JT)$.

12.6.3 Simultaneous Solution of the Differenced, Linearized Mass and Momentum Equations

If the discretized mass and momentum equations derived in Section 12.6.2 (Eq. (12.6-45) and Eq. (12.6-132)) are applied to each axial segment in the bubble, the result is a set of simultaneous linear equations which have as independent variables only the changes in vapor pressure and in mass flow rate. To this set must be added the equations at the upper and lower bubble interfaces (Eq. (12.6-79), Eq. (12.6-87), Eq. (12.6-43), and Eq. (12.6-153)), giving a set of $2\left(J2 - J1\right)$ equations in $2\left(J2 - J1 + 1\right)$ unknowns. To complete the set, a boundary condition must be imposed at both interfaces. This requirement is satisfied by expressing the interface mass flow rate as

$$W_1^i\left(JT\right) = v_1^i\left(JT\right)\rho_1^i\left(JT\right)A_c(JX) \tag{12.6-160}$$

which can be differenced to give

$$\Delta W^i\left(JT\right) = \left(v_1^i\left(JT\right)\Delta\rho^i\left(JT\right) + \rho_1^i\left(JT\right)\Delta v^i\left(JT\right)\right)A_c(JX) \tag{12.6-161}$$

The index JT is $J1$ for the lower interface and $J2$ for the upper interface, while JX is $J1$ for the lower interface and $J2-1$ at the upper one. Using the expressions derived earlier for $\Delta\rho^i\left(JT\right)$ and $\Delta v^i\left(JT\right)$ (Eq. (12.6-37) and Eq. (12.6-62)) produces

$$\begin{aligned}\Delta W^i\left(JT\right) = \left(v_1^i\left(JT\right)\frac{d\rho^i}{dp_{JT}} + \rho_1^i\left(JT\right)\frac{dv^i}{dp_{JT}}\right)A_c\left(JX\right)\Delta p^i\left(JT\right) \\ +\Delta v_0^i\left(JT\right)A_c\left(JX\right)\rho_1^i(JT)\end{aligned} \tag{12.6-162}$$

which can be rearranged to give

$$-\left(v_1^i\left(JT\right)\frac{d\rho^i}{dp_{JT}}+\rho_1^i\left(JT\right)\frac{dv^i}{dp_{JT}}\right)A_c\left(JX\right)\Delta p^i\left(JT\right)+\Delta W^i\left(JT\right)$$
$$=\rho_1^i\left(JT\right)\Delta v_0^i\left(JT\right)A_c(JX) \tag{12.6-163}$$

Define

$$a_{1,JX}=-\left(v_1^i\left(JT\right)\frac{d\rho^i}{dp_{JT}}+\rho_1^i\left(JT\right)\frac{dv^i}{dp_{JT}}\right)A_c(JX) \tag{12.6-164}$$

$$a_{2,JX}=1.0 \tag{12.6-165}$$

and

$$d_{JX}=\rho_1^i\left(JT\right)\Delta v_0^i\left(JT\right)A_c(JX) \tag{12.6-166}$$

Then Eq. (12.6-163) becomes

$$a_{1,JX}\Delta p^i\left(JT\right)+a_{2,JX}\Delta W^i\left(JT\right)=d_{JX} \tag{12.6-167}$$

If Eq. (12.6-167) is applied to both upper and lower interfaces and the resulting two equations are added to the set of mass and momentum equations, a set of $2\left(J2-J1+1\right)$ equations in $2\left(J2-J1+1\right)$ unknowns will result. This set can be written in matrix form as

$$\mathbf{AX}=\mathbf{B} \tag{12.6-168}$$

where the form of matrix **A** is given by Eq. (12.6-169) and the **X** and **B** vectors are given by Eq. (12.6-170) and Eq. (12.6-171).

$$\mathbf{A}=\begin{bmatrix}
a_{1,J1} & a_{2,J1} & 0 & 0 & 0 & 0 & \cdots & 0 & 0 & 0 & 0\\
\widetilde{b}_{1,J1} & \widetilde{b}_{2,J1} & \widetilde{b}_{3,J1} & \widetilde{b}_{4,J1} & 0 & 0 & \cdots & 0 & 0 & 0 & 0\\
c_{1,J1} & c_{2,J1} & c_{3,J1} & c_{4,J1} & 0 & 0 & \cdots & 0 & 0 & 0 & 0\\
0 & 0 & b_{1,J1+1} & b_{2,J1+1} & b_{3,J1+1} & b_{4,J1+1} & \cdots & 0 & 0 & 0 & 0\\
0 & 0 & c_{1,J1+1} & c_{2,J1+1} & c_{3,J1+1} & c_{4,J1+1} & \cdots & 0 & 0 & 0 & 0\\
 & & & & \ddots & \ddots & \ddots & \ddots & \ddots & & \\
0 & 0 & 0 & 0 & 0 & 0 & \cdots & \widetilde{b}_{1,J2-1} & \widetilde{b}_{2,J2-1} & \widetilde{b}_{3,J2-1} & \widetilde{b}_{4,J2-1}\\
0 & 0 & 0 & 0 & 0 & 0 & \cdots & c_{1,J2-1} & c_{2,J2-1} & c_{3,J2-1} & c_{4,J2-1}\\
0 & 0 & 0 & 0 & 0 & 0 & \cdots & 0 & 0 & a_{1,J2-1} & a_{2,J2-1}
\end{bmatrix} \tag{12.6-169}$$

$$\mathbf{X}=\begin{bmatrix}
\Delta p^i(J1)\\
\Delta W^i(J1)\\
\Delta p(J1+1)\\
\Delta W(J1+1)\\
\vdots\\
\Delta p(J2-1)\\
\Delta W(J2-1)\\
\Delta p^i(J2)\\
\Delta W^i(J2)
\end{bmatrix} \tag{12.6-170}$$

and

$$B = \begin{bmatrix} d_{J1} \\ \widetilde{g}_{J1} \\ h_{J1} \\ g_{J1+1} \\ h_{J1+1} \\ \vdots \\ \widetilde{g}_{J2-1} \\ h_{J2-1} \\ d_{J2-1} \end{bmatrix} \tag{12.6-171}$$

This matrix equation can be solved using Gaussian elimination to give the changes in vapor pressure and mass flow rate at all nodes in the bubble. Section 12.16 presents a step-by-step description of how Eq. (12.6-168) is solved using Gaussian elimination. Once the new vapor pressures are known, the vapor temperatures and all thermodynamic sodium properties can be calculated.

12.7 Voiding Due to Gas Release from Failed Fuel Pins

The boiling model in SAS4A/SASSYS-1 has been modified to account for voiding due to release of plenum gas from failed fuel pins. The main purpose of this modification is to address the question of whether pin failure in a transient that would otherwise not lead to boiling recovery would be prevented. Another case that can be addressed with this model is one in which pin failure occurs after the onset of boiling and increases the severity of the voiding. DEFORM-5 predicts the time and location of pin failure, or the user can specify the time and location of the pin failure. The gas release then provides a fictitious source of noncondensing vapor for the boiling model. An adjustable rip area and orifice coefficient determine the rate at which gas is released from the pins. Reduction in the plenum pressure is calculated as the gas flows out the rip. The gas fraction in each bubble in the coolant channel is calculated. The presence of gas in a bubble has two effects: it reduces the condensation coefficient and it increases the vapor friction factor. As the gas bubbles out the top of the subassembly a smooth transition is made to normal boiling. Three pin failure groups are used to account for incoherence. Each group represents a fraction of the pins. A separate life fraction is used by DEFORM-5 to predict failure for each pin group. Each pin group has a separate plenum gas pressure and temperature and a separate flow rate out the rupture.

For this model the rate of gas flow from the plenums to the coolant channel is calculated as

$$w_g = \sum_m f_m A^* \sqrt{\frac{2\left(p_{pm} - p_c\right)\overline{\rho}_{gm}}{K_g}} \tag{12.7-1}$$

where

m = pin group number

p_{pm} = plenum pressure for pin group m

p_c = pressure in the coolant channel

A^* = rip area

K_g = rip orifice coefficient

$$\overline{\rho}_{gm} = \frac{\rho_{gm} + \rho_{gc}}{2} \tag{12.7-2}$$

f_m = fraction of the pins in group m

ρ_{pm} = gas density in the plenum

ρ_{gc} = gas density in the coolant channel

Note that for $p_{pm} \gg p_c$ and $K_g \quad 2$, Eq. (12.7-1) reduces to the expression for choked flow through as orifice, whereas for small pressure differences the equation reduces to the ordinary orifice flow expression.

Also, for a perfect gas:

$$p_{pm} V_p = m_{pm} R_g T_{pm} \tag{12.7-3}$$

or

$$p_{pm} = \rho_{pm} R_g T_{pm} \tag{12.7-4}$$

and

$$\rho_{gc} = \frac{p_c}{R_g T_{gc}} \tag{12.7-5}$$

where

p_c = pressure in the coolant channel near the rip

T_{gc} = gas temperature as it enters the coolant channel

p_{pm} = plenum gas pressure

R_g = gas constant

T_{pm} = gas temperature in the plenum

The gas flow at the end of the time step is calculated as

$$w_g\,(t + \Delta t) = \sum_m f_m (w_{gm1} + \Delta p_v w_{gm3}) \tag{12.7-6}$$

where

t = time at beginning of step

Δt = step size

Δp_v = change in coolant channel pressure during the step

$$w_{gm1} = A^* \sqrt{\frac{[p_{pm}\,(t) - p_c(t)]\,2\overline{\rho}_{gm}(t)}{K_g}} \tag{12.7-7}$$

$$w_{gm3} = \frac{A^* \overline{\rho}_{gm}}{K_g \sqrt{\frac{[p_{pm}(t) - p_c(t)] 2\overline{\rho}_{gm}(t)}{K_g}}} \tag{12.7-8}$$

The gas that comes into the coolant channel is converted into enough vapor to have the same pressure-volume product. Thus

$$p_g \frac{w_g}{\rho_{gc}} = p_v \frac{w_v}{\rho_v} \tag{12.7-9}$$

where

w_v = vapor source

p_v = vapor pressure

ρ_v = vapor density

A conversion factor, γ_g between gas and vapor is defined as

$$w_g = \gamma_g w_v \tag{12.7-10}$$

Then

$$\gamma_g = \frac{p_v/\rho_v}{p_g/\rho_g} = \frac{p_v/\rho_v}{R_g/T_{gc}} \tag{12.7-11}$$

The temperature used for T_{gc}, the temperature of the gas entering the coolant channel, is the fuel surface temperature at the rupture node.

The treatment of the gas after it enters the coolant channel depends on whether it is entering a small, nonpressure gradient bubble on a larger, pressure gradient bubble. In a small, nonpressure gradient bubble the gas adds to the heat flow, Q_{es}, from cladding and structure to the vapor:

$$\begin{aligned} Q_{\text{es}}(t+\Delta t) = I_{e1} + \delta I_{e1g} + \Delta T_v \left(I_{e2} + \delta I_{e2g}\right) \\ + I_{e3}\Delta Z'(K,2) + I_{e4}\Delta Z'(K,1) \end{aligned} \tag{12.7-12}$$

where

ΔT_v = change in vapor temperature for the time

$\Delta Z'(K,2)$ = change in lower bubble interface position due to change in bubble pressure

$\Delta Z'(K,1)$ = change in lower bubble interface position due to change in bubble pressure

K = bubble number

$I_{e1}, I_{e2}, I_{e3}, I_{e4}$ = contribution from normal vapor sources

$\delta I_{e1g}, \delta_I e2g$ = gas source contributions

$$\delta I_{e1g} = \frac{\lambda_v}{\gamma_g} A^* \sum_m f_m \sqrt{\frac{\left[p_{pm}(t) - p_c(t)\right] 2\overline{\rho}_{gm}(t)}{K_g}} \tag{12.7-13}$$

$$\delta I_{e2g} = \frac{\lambda_v}{\delta_g} A^* \frac{\partial p_v}{\partial T_v} \sum_m f_m \frac{\overline{\rho}_{gm}(t)}{K_g} \sqrt{\frac{K_g}{2\overline{\rho}_{gm}(t)\left[p_{pm}(t) - p_c(t)\right]}} \tag{12.7-14}$$

λ_v = heat of vaporization

For a larger, pressure gradient bubble Eq. (12.6-45) gives a mass equation of the form

$$C_{1,J}\Delta p(J) + C_{2,J}\Delta W(J) + C_{3,J}\Delta p(J+1) + C_{4,J}\Delta W(J+1) = h_J \tag{12.7-15}$$

For the mode J at which the pin rupture occurs the gas adds contributions to $C_{1,J}, C_{3,J}$ and h_J:

$$h_J \rightarrow h_J + \sum_m \frac{\Delta t f_m}{\lambda_g} w_{gm1} \tag{12.7-16}$$

$$C_{1,J} \rightarrow C_{1,J} + d_2 \tag{12.7-17}$$

$$C_{3,J} \rightarrow C_{3,J} + d_2 \tag{12.7-18}$$

$$d_2 = \sum_m \frac{\Delta t}{2\gamma_g} f_m w_{gm3} \tag{12.7-19}$$

The gas plenum calculations for a time step are done in two parts. First the effects of gas release are calculated. Then the heat flow from the cladding to the gas is accounted for as before.

The gas mass lost from the plenum in pin group m during a time step is calculated using

$$m_{pgm}\left(t+\Delta t\right) = m_{pgm}\left(t\right) - \Delta t\left(w_{gm1} + \frac{\Delta p_v}{2} w_{gm3}\right) \tag{12.7-20}$$

where

m_{pgm} = gas mass in plenum for group m

Then the remaining gas in the plenum is expanded adiabatically:

$$p_{pm}\left(t+\Delta t\right) = p_{pm}\left(t\right)\left(\frac{m_{pgm}\left(t+\Delta t\right)}{m_{pgm}\left(t\right)}\right)^{\gamma} \tag{12.7-21}$$

where

γ = gas specific heat at constant pressure/specific heat at constant volume

The gas temperature, T_{pgm}, for the first part of the step is calculated as:

$$T_{pgm}\left(t+\Delta t\right) = \frac{p_{pm}\left(t+\Delta t\right) V_p}{m_{pgm}\left(t+\Delta t\right) R_g} \tag{12.7-22}$$

where

V_p = gas plenum volume

Then heat flow from the cladding is used to recalculate $T_{pgm}\left(t+\Delta t\right)$ and $p_{pm}\left(\ t+\Delta t\right)$.

The amount of gas in each bubble is kept track of, and the gas fraction in a bubble affects the condensation heat transfer coefficient and the friction factor. The mass of gas, m_{gbk}, in bubble k is calculated using

$$m_{gbk}\left(t+\Delta t\right) = m_{gbk}\left(t\right) + \frac{\Delta t}{\gamma_g}\sum_m \left(w_{gm1} + \frac{\Delta p_v}{2} w_{gm3}\right) \tag{12.7-23}$$

The total mass, m_{tk}, in each bubble is calculated as

$$m_{tk} = \int \rho_v\left(z\right) A_c\left(z\right) dz \tag{12.7-24}$$

where

A_c = coolant flow area

The gas fraction, f_{gk}, in bubble k is then defined as

$$f_{gk} = \frac{m_{gbk}}{m_{tk}} \tag{12.7-25}$$

Gas stays in bubble until it blows out the top of the subassembly. When a vapor bubble extends far enough out the top of the subassembly, part of the bubble is assumed to break away and go up into the outlet plenum. If the bubble upper position, z_{iuk}, exceeds the top of subassembly, z_{tsa}, by an amount Δz_{bu} (e.g. $z_{iuk} \geq z_{tsa} + \Delta z_{bu}$) then the part of the bubble above $z_{tsa} + \Delta z_{bl}$ is broken off from the bubble. When this happens m_{gbk} is decreased by the amount of gas in the part broken off:

$$m_{gbk} \rightarrow m_{gbk} - \Delta m_g \tag{12.7-26}$$

where

$$\Delta m_g = f_{gk} A_c \rho_v \left(z_{iuk} - z_{tsa} - \Delta z_{bl} \right) \tag{12.7-27}$$

Then the upper interface of the bubble is moved down to $z_{tsa} + \Delta z_{bl}$:

$$z_{iuk} \rightarrow z_{tsa} + \Delta z_{bl} \tag{12.7-28}$$

Also, each time that the top part of a bubble is broken off, the velocity of the upper interface is cut in half. The condensation coefficient, h_c, in a bubble containing gas is calculated as

$$h_c = \frac{h_{co}}{1 + 100,000 f_{gk}} \tag{12.7-29}$$

where

h_{co} = normal vapor condensation coefficient.

Gas streaming in a bubble will have a different friction pressure drop than sodium vapor moving at the same velocity, so in this model the vapor friction factor is adjusted when gas is present in a bubble. The friction pressure drop, Δp_f, due to vapor streaming in the coolant channel is

$$\Delta p_f = \frac{L}{D_h} f \rho_v v^2 \tag{12.7-30}$$

where

f = friction factor

v = vapor velocity

L = length

D_h = hydraulic diameter

When gas is turned into fictitious vapor in the coolant channel, its pressure-volume product is conserved; so its velocity should be approximately correct. On the other hand, fission product gas has a higher density than sodium vapor, so the friction pressure drop will not be correct unless the density ratio is accounted for. In this model, the friction factor is adjusted to account for the density ratio:

$$f = f_0 \left(\frac{M_{\text{wg}}}{M_{\text{wNa}}} f_{\text{gk}} + 1 - f_{\text{gk}} \right) \tag{12.7-31}$$

where

f_0 = normal friction factor

M_{wg} = molecular weight of the gas

M_{wNa} = molecular weight of sodium, taken to be 46

12.8 Time Step Controls

In order to operate the voiding model as efficiently and accurately as possible, provision is made for varying the coolant time-step size. This allows a large step size to be used in portions of the transient when no rapid changes are taking place, with a transfer to small time steps when changes occur more rapidly. The size of the step is bounded from below by a user-input minimum value (variable **TPDMIN**) and is limited above by the user-input maximum heat-transfer (**DTMXB**), primary-loop (**DTPBOI**), and main (**DTMMXB**) time step sizes. Within this rather broad range, the actual step size is determined as the largest value which will satisfy several limiting criteria on other quantities involved in the calculation. These criteria require the time-step size to be small enough so that all of the following are satisfied:

1. No bubble which is decreasing in size will shrink to less than half its size from the start to the end of the time step. This prevents collapsing bubbles from disappearing in the middle of the time step and allows them to be removed by the bubble-size criterion at a point in the coding at which the loss of the bubble may be properly accounted for.
2. Neither the highest not the lowest liquid slug is ejected from the subassembly over the span of the time step. As in item 1, such an event must be recorded at the end of a time step in a specific section of the coding.
3. Neither the liquid-sodium nor sodium-vapor temperature anywhere in the channel changes by more than a user-input amount (**DTLMAX** and **DTVMAX**, respectively) over the time step.
4. The liquid-slug mass flow rate in any slug does not change by more than thirty percent over the time step.
5. Neither the highest nor the lowest vapor bubble is ejected from the subassembly over the span of the time step.
6. No vapor-liquid interface travels more than a user-supplied distance (**DZIMAX**) over the time step.
7. No vapor-liquid interface crosses more than one axial mesh segment boundary during the time step.
8. No liquid slug shrinks to less than a minimum length (**SLMIN**) over the time step.

In addition to satisfying these criteria, the time-step selection must obey the following rules:

1. No time step can be less than 10^{-7} seconds.
2. The new time step cannot be more than four times as large as the old step, even if it satisfies the eight criteria listed above.
3. The coolant time step cannot extend past the end of the heat-transfer or primary-loop time steps.

Not that it is possible that the time step which satisfied the eight limiting criteria may be less than 10^{-7} seconds long, which violates the first rule shown above. In this case, the time step is simply set to 10^{-7} seconds, even though one or more of the criteria may not be satisfied.

The size of the new time step is set in subroutine TSC9 at the end of the voiding calculation over the old time step. This subroutine determines the time step which will satisfy the rules and criteria just discussed. In addition, checks are made on several of the criteria during the course of the calculation; if the time step fails to meet any of these checks, it is reset at the point at which the check was made and the calculation for that step is started over again.

A special case exists when boiling is initiated in any channel. In this instance, the coding will iterate to try to determine the exact time of voiding onset, and so the time step must be adjusted repeatedly so as to locate the beginning of boiling exactly. The time-step adjustment is made from a linear extrapolation based on the amount of liquid superheat.

12.9 Interaction with Other Models

As mentioned in the introduction to this chapter, the voiding model interacts directly with several other major modules in SASSYS-1. This section will provide a more detailed discussion of these interactions.

The voiding model first makes use of information from another module by accessing initialization data from the prevoiding calculation. Quantities such as liquid sodium temperatures, pressures, and mass flow rates are computed in the prevoiding module and passed to the voiding model coding, once the prevoiding calculation has predicted that initiation of voiding is imminent. The voiding model then begins interacting with the transient heat-transfer module at the end of each heat-transfer time step. The heat-transfer calculation provides the cladding and structure axial temperature profiles required by the voiding model (T_e and T_s, respectively). The

voiding model estimates these temperatures at the end of the new heat-transfer time step by extrapolating the cladding and structure temperatures from the ends of the previous two heat-transfer time steps. It then computes the advanced time cladding and structure temperatures for each coolant time step by interpolating between the temperatures at the beginning and the end of the heat-transfer time step. During the course of the heat-transfer time step, the voiding model computes the time-integrated heat fluxes from the cladding (E_e) and the structure (E_s) and supplies them to the transient heat-transfer calculation at the end of the heat-transfer time step.

Reactivity feedback due to coolant voiding is provided by the voiding model through calculation of a spatially smeared liquid sodium density for each axial mesh segment. This density is passed to the point kinetics module, where it is used to compute the sodium void fraction in each segment. The reactivity change due to voiding is then computed from

$$\rho_c = \sum_I \sum_j (\rho_c)_{jI} \, \alpha_{jI} \tag{12.9-1}$$

where

$(\rho_c)_{jI}$ = reactivity worth of sodium in axial segment j of channel I (input through variable **VOIDRA**)

and

α_{jI} = average sodium void fraction in segment j of channel I.

Although the point kinetics module does not feed any information back directly to the voiding model, it does supply the transient heat-transfer module with information about the change in power, which in turn affects the computation of the cladding temperatures, and so the point kinetics model does indirectly influence the voiding model.

12.10 Detailed Flow Description

The logic of the voiding model coding is displayed in the flow diagrams of Figure 12.10.1. The flowchart is broken up into several sections: the main flow diagram, which outlines the structure of the driver subroutine TSBOIL, and several auxiliary diagrams which give more detail about larger subroutines called by TSBOIL. A brief description of the function of each subroutine used in the voiding model is provided in Table 12.10.1 immediately following Figure 12.10.1.

The flowchart may be easier to follow if it is studied in conjunction with a more detailed description of the coding logic. The driver subroutine TSBOIL is first called by TSTHRM once the prevoiding model has predicted that initiation of voiding in the channel is imminent. On this first call, TSBOIL uses subroutines CHIN and TSILLB to initialize variables needed in the voiding model calculation from information generated in the prevoiding model. Subroutine TSCSET is then called to update voiding-related variables to the current coolant time step. Temperatures and liquid-sodium material properties are extrapolated to the end of the time step in subroutine TSC2, which, together with MOMEN and TSC21, also solves the liquid slug momentum equation for a preliminary value of the liquid mass flow rate in the channel at the end of the time step. The code then branches to subroutine TSOV45. This subroutine completes the liquid slug calculations by calling TSC5 to compute the final value of the mass flow rate, TSC6 to determine the liquid sodium temperatures along the channel, and TSC7 to find the axial pressure distribution and saturation temperatures and to determine whether the exact time of voiding initiation has been calculated or whether it is necessary to iterate on the start of voiding. If iteration is necessary, the program adjusts the time step for a new estimate of the time of voiding initiation and returns to TSC2. This process is repeated until the time of the start of boiling has been found. At this point, TSOV45 calls TSC8, which uses TSCBUB, TSC85, and TSC86 to initialize the variables necessary to account properly in the voiding calculation for the presence of the newly formed bubble. Next, the increments in the heat fluxes and the heat-transfer coefficients and the temperatures of the structure, reflectors, plenum gas, and

plenum cladding are computed through TSCA, TSCA1, and TSCA2. The new time-step size is set in TSC9, and the updated heat fluxes and heat-transfer coefficients are determined in TSCA. Before returning control to TSBOIL, TSOV45 checks to see if cladding motion has started, and, if it has, it calls TSCLD2. The coding then returns to TSBOIL. A check is made to see if the end of the primary-loop time step has been reached, and if it has, the PRIMAR-4 model is called through PRIMUP and the inlet and outlet temperatures are updated from TSCMP1. Tests are then made on error flogs and on whether the heat-transfer time step has been completed; if it has, a new heat-transfer time step is calculated in TSHTRV. If the appropriate debug flag is set, a printout of some of the voiding model variables is created by NODEPR; otherwise, the coolant time step is over, and control is returned to TSTHRM.

On subsequent calls to TSBOIL, the code branches immediately to TSCSET to update the appropriate variables. Subroutine TSC2 is then called, and then the film thicknesses on the cladding and structure in any voided regions, as well as some heat-transfer coefficient and heat-flux information, are computed using TSC3 and TSC31. The subroutine T4A3D is then called to perform the voiding model calculation. This subroutine first checks for any bubbles that are small enough to use the uniform-vapor-pressure model and calls TSC43A to compute the vapor temperatures and pressures in these bubbles. The pressure-gradient mode is applied by T4A3D to any remaining bubbles, with subroutine T41A3D used to set up the matrix Eq. (12.6-168) and T42A3D responsible for applying the Gaussian elimination procedure to solve the equation. Once the voiding calculation is completed, TSOV45 is called. This subroutine and the subroutines it calls perform the operations described above; in addition, TSC5 computes the liquid-vapor interface positions and TSC8, with the help of TSC82, TSC83, TSC84, TSC85, and TSC86, takes care of all liquid-slug and vapor-bubble accounting. Phenomena such as bubble formation and collapse; liquid-slug disappearance, expulsion, and reentry; and vapor-bubble combination are treated in these subroutines. Subroutine TSC9 not only computes a new time step but also performs a series of checks to see if the current time step should be cut back and the calculation redone; if this is the case, the step size is cut, the coding branches back to the middle of TSCSET for extrapolation to the end of the revised time step of the required variables, and the calculation is repeated. Otherwise, TSOV45 returns control to TSBOIL, and the remainder of the computation proceeds as above.

Table 12.10.1: Alphabetical Listing of Voiding Model Subroutines with Descriptions

Routine	Description
CHIN	Calculates the ratio of the fuel pin outer perimeter to the coolant area. Called by TSBOIL.
DTHFND	Calculates new heat-transfer time step. Called by TSBOIL.
MOMEN	Computes the terms in the momentum equation for each segment of each liquid slug. Called by TSC2.
NODEPR	Gives printout of all temperatures. Called by TSBOIL.
ORUNYO	Updates data for PRIMAR-4. Called TSBOIL.
TSBOIL	Drives the two-phase calculation. Also, forms a new gas bubble, if necessary, when pin rupture occurs, and reduces the time step when approaching rupture. Called for TSTHRM each coolant time step.
TSCA	Comprises two independent sections. First section calls TSCA1 and TSCA2. Second section computes film heat-transfer coefficients and heat fluxes and sets cladding temperatures in the fuel pin for the next time step. Reduces the gas mass in a bubble when part of the bubble is cut off after blowing out the top of the subassembly. Called by TSOV45.
TSCA1	Calculates increments to the integrated heat fluxes from the cladding to the coolant and from the structure to the coolant. Also calculates increments to the film heat-transfer coefficients and the average sodium density. Called by TSCA.

continues on next page

Table 12.10.1 – continued from previous page

Routine	Description
TSCA2	Calculates temperatures of structure, reflectors, cladding in the plenum, and gas in the plenum. Also, calculates mass of gas in the plenum and in a bubble. Called by TSCA.
TSCBUB	Creates a new vapor bubble. Also, sets the initial gas mass in a bubble to zero. Called by TSC8 and TSC82.
TSCC	Calculates film motion (not currently used). Called by TSOV45.
TSCLD1	Driver subroutine for cladding motion. Called by T4A3D.
TSCLD2	Performs moving cladding velocity calculation. Called by TSO45.
TSCMP1	Updates inlet and outlet pressures. Called by TSBOIL.
TSCSET	Initializes coolant variables and gas release variables each coolant time step. Called by TSBOIL.
TSC2	Extrapolates coolant, cladding, and structure temperatures; computes density, thermal conductivity, heat capacity, and viscosity at the extrapolated time; and computes the change in coolant flow rate, neglecting the pressure change with time at the interfaces. Called by TSBOIL.
TSC21	Computes the terms in the momentum equation for each fully liquid segment of each coolant slug. Called by MOMEN.
TSC3	Computes coolant heat transfer coefficient and film thicknesses. Also calculates heat flow and gas source to uniform pressure bubbles. Adjusts condensation coefficient when gas is present. Called by TSBOIL.
TSC31	Function used by TSC3 to calculate liquid film heat transfer coefficients in voided regions.
TSC43A	Calculates vapor temperatures and pressures in uniform pressure bubbles. Called by T4A3D.
TSC5	Calculates liquid slug flow rates and liquid-vapor interface positions after the first bubble is formed. Called by TSOV45.
TSC6	Calculates temperatures in liquid slugs. Called by TSOV45.
TSC7	Computes pressure and saturation temperatures in liquid slugs and performs the iteration to specify the time at which boiling starts. Called by TSOV45.
TSC8	Performs accounting for bubble collapse, bubble formation, liquid slug expulsion and reentry, and combination of bubbles upon disappearance of a liquid slug. Called by TSOV45.
TSC82	Checks for formation of a new bubble (other than the first bubble formed in the subassembly). Called by TSC8.
TSC83	Calculates liquid slug expulsion and reentry. Called by TSC8.
TSC84	Calculates bubble collapse. Called by TSOV45.
TSC85	Performs vapor bubble accounting. Called by TSC8, TSCBUB, and TSC84.
TSC86	Performs liquid slug accounting. Called by TSC8, TSCBUB, and TSC84.
TSC87	Keeps track of which bubble is in contact with the rupture if gas is being released. Called from TSC8.
TSC9	Computes next coolant time step size. Also evaluates and resets the current time step if the step size is too large. Called by TSOV45.
TSHTRV	Transient state heat transfer. Called by TSBOIL.
TSILLB	Initializes cladding temperatures, velocities at interfaces, and interface positions. Called by TSBOIL only for the first coolant time step.
TSOV45	Drives the calculation of the pressure, temperature, and flow rates in the liquid slugs and the bubble accounting. Also calculates the gas fraction in each bubble. Called by TSBOIL.

continues on next page

Table 12.10.1 – continued from previous page

Routine	Description
T4A3D	Separates pressure drop and uniform-pressure bubble calculations and initializes variables for calculation of vapor pressures and mass flow rates. Called by TSBOIL.
T41A3D	Sets up matrix for pressure-drop bubble calculations. Also, for the gas release model, calculates the gas source into pressure gradient bubbles and adjusts friction factors. Called by T4A3D.
T42A3D	Calculates solution of simultaneous equations by Gaussian elimination to solve for changes in pressures and mass flow rates in pressure-drop bubbles. Called by T4A3D.

12.11 Input Description

Table 12.11.1 lists the input information that is needed in order to run the voiding model section of SAS4A/SASSYS-1. This information is of two types: that used directly in the voiding calculation and that used by models which provide necessary input to the voiding model. This listing is a subset of the full input listing given in Section 12.16. To simplify the discussion of the information displayed in Table 12.11.1, the input variables will not be examined in the order presented in the table, but rather, they will be presented in groups of data by category, i.e., all variables pertaining to materials properties will be discussed together, all which give problem geometry information will be grouped together, etc. This discussion should be helpful to users trying to assemble input decks for running their own problems.

The variables which provide information on materials properties will be described first. Most materials-property information, other than that for sodium, is provided through input rather than being fixed within the code in order to allow greater flexibility in the use of SASSYS-1. In most cases, this information is input in the form of tables; however, in the case of the solid fuel density and fuel thermal conductivity, either tables or coefficients for functional relationships can be entered. The choice between these two options is made on a channel-by-channel basis and is flagged by the variable **IRHOK**; a positive value of **IRHOK** indicates that the functional form has been selected, and a non-positive value means that the tabular format has been chosen. The variable **COEFDS** contains the coefficients for the fuel density relationship as a function of temperature, while **RHOTAB** contains a table of fuel density as a function of temperature **RHOTEM**. Similarly, the coefficient for the thermal conductivity function can be read into **COEFK**, while a table of thermal conductivity values can be used in **XKTAB** together with a table of temperatures in **XKTEM**. In addition, the specific heat of the fuel is read into the table **CPFTAB** as a function of temperature **CPFTEM**.

Table 12.11.1: List of Input Variables Needed by the Voiding Model

Equation Variable	Reference Equation	Input Variable	Suggested Value	External Reference
-	-	**NCHAN**	-	-
-	-	**IFUEL1**	-	-
-	-	**ICLAD1**	-	-
-	-	**IPLUP**	-	-
-	-	**ITKEL**	-	-
-	-	**IPOWER**	-	-
-	-	**IPOWOP**	-	-
-	-	**MAXSTP**	-	-
-	-	**IPO**	-	-
-	-	**IPOBOI**	-	-

continues on next page

Table 12.11.1 – continued from previous page

Equation Variable	Reference Equation	Input Variable	Suggested Value	External Reference
-	-	IBLPRT	-	-
-	-	NPREAT	-	-
-	-	NPRES	-	-
-	-	IFLOW	-	-
-	-	NT0TAB	-	
-	-	DT0	-	-
-	-	DTMXB	-	-
-	-	TIMAX	-	-
-	-	DTFUEL	-	-
-	-	DTCLAD	-	-
-	-	DTMMXB	-	-
-	-	DPWMAX	-	-
	-	POW	-	-
	-	PREATB	-	-
	-	PREATM	-	-
	-	FRPR	-	-
	-	FRFLOW	-	-
ρ_f	-	COEFDS	11.08×10^3 2.04×10^{-5} 8.7×10^{-9}	-
k_f	-	COEFK	2.1, 2.88×10^{-3} 2.52×10^{-5} 5.83×10^{-10} 5.75×10^{-2} 5.03×10^{-4} 2.91×10^{-11}	-
k_c	-	EXKTB	-	-
-	-	EXKTM	-	-
ρ_f	-	RHOTAB	-	-
-	-	RHOTEM	-	-
-	-	TR	300	-
k_f	-	XKTAB	-	-
-	-	XKTEM	-	-
C_f	-	CPFTAB	-	-
-	-	CPFTEM	-	-
-	-	TFSOL	-	-
-	-	TFLIQ	-	-
-	-	UFMELT	-	-
-	-	TESOL	-	-
-	-	TELIQ	-	-
-	-	UEMELT	-	-
-	-	CPCTAB	-	-
-	-	CPCTEM	-	-
$(\rho)_e$	-	CROETB	-	-
-	-	CROETM	-	-
C_e	-	CE	690	-
γ	Eq. (12.7-21)	GAMGAS	1.3 - 1.6	-
P_x	-	PX	-	-
-	-	PDEC	-	-

continues on next page

Table 12.11.1 – continued from previous page

Equation Variable	Reference Equation	Input Variable	Suggested Value	External Reference
-	-	PDEC1	-	-
-	-	PDEC2	-	-
-	-	PRETAB	-	-
-	-	PRETME	-	-
T_O	-	T0TAB	-	-
-	-	T0TME	-	-
-	-	ZPLENL	-	-
-	-	ZPLENU	-	-
ΔZ_{bl}	Eq. (12.7-28)	DZBCGL	.1	-
ΔZ_{bu}	-	DZBCGU	.25	-
-	-	IDBUGV	-	-
-	-	IERSTP	-	-
-	-	IRHOK	-	-
-	-	NPLN	-	-
-	-	NREFB	-	-
-	-	NREFT	-	-
-	-	NZNODE	-	-
-	-	NT	-	-
-	-	IFUELV	-	-
-	-	IFUELB	-	-
-	-	ICLADV	-	-
-	-	IHGAP	-	-
-	-	NPIN	-	-
-	-	NSUBAS	-	-
-	-	MZUB	-	-
-	-	MZLB	-	-
-	-	IHEX	-	-
-	-	ISSFUE	-	-
-	-	ILAG	-	-
-	-	ICLADB	-	-
-	-	MFAIL	-	-
-	-	IFAIL	-	-
-	-	JFAIL	-	-
-	-	IDBFLG	-	-
-	-	IDBSTP	-	-
-	-	IEQMAS	-	-
-	-	IBLPRN	-	-
-	-	IDBGBL	-	-
-	-	IDBLST	-	-
-	-	ISSFU2	-	-
-	-	IFILM	-	-
-	-	NZONF	-	-
-	-	IFUELI	-	-
-	-	NODSUM	-	-
-	-	IGASRL	0 or 1	-
-	-	IGRLTM	0 or 1	-
A_c	Eq. (12.2-1)	ACCZ	-	-
Δz	Eq. (12.2-4)	AXHI	-	-

continues on next page

Table 12.11.1 – continued from previous page

Equation Variable	Reference Equation	Input Variable	Suggested Value	External Reference
D_h	Eq. (12.2-7)	DHZ	-	-
-	-	DSTIZ	-	-
-	-	DSTOZ	-	-
-	-	PLENL	-	-
-	-	RBR	-	-
-	-	RER	-	-
-	-	RBRPL	-	-
-	-	RERPL	-	-
-	-	RINFP	-	-
-	-	ROUTFP	-	-
-	-	ZONEL	-	-
-	-	SRFSTZ	-	-
-	-	AREAPC	-	-
-	-	DRFO	-	-
-	-	RBR0	-	-
-	-	RER0	-	-
p_e	Eq. (12.3-2)	SER	-	-
-	-	DRFI	-	-
-	-	VFC	-	-
γ_s	-	GAMSS	-	-
γ_c	-	GAMTNC	-	-
γ_e	-	GAMTNE	-	-
-	-	PSHAPE	-	-
-	-	PSHAPR	-	-
-	-	AHBPAR	-	-
-	-	BHBPAR	-	-
-	-	CHBPAR	-	-
-	-	HBMAX	-	-
-	-	HBMIN	-	-
-	-	HBPAR	-	-
k_s	-	XKSTIZ	-	-
k_s	-	XKSTOZ	-	-
-	-	DEL	-	-
-	-	P0GAS	-	-
k_f	-	XKRF	-	-
ρ_c	-	DENSS	-	-
$(\rho c)_s$	-	RHOCSI	-	-
$(\rho c)_s$	-	RHOCSO	-	-
$(\rho c)_r$	-	RHOCR	-	-
$(\rho c)_g$	-	RHOCG	-	-
R_g	-	RG	-	-
A_{fr}	Eq. (12.2-2)	AFR	0.1875	5-22
b_{fr}	Eq. (12.2-2)	BFR	-0.2	5-22
-	-	C1	-	5-22
-	-	C2	-	5-22
-	-	C3	-	5-22
-	-	DWMAX	0.2	-
w	-	W0	-	-

continues on next page

Table 12.11.1 – continued from previous page

Equation Variable	Reference Equation	Input Variable	Suggested Value	External Reference
K_{or}	Eq. (12.2-9)	XKORI	-	5-23
$(\Delta z_i/A)_b$	Eq. (3.9-9)	DZIAB	$(D_h/2A_c)$	-
$(\Delta z_i/A)_t$	Eq. (3.9-9)	DZIAT	$(D_h/2A_c)$	-
θ_1	Eq. (12.2-17)	THETA1	-	-
θ_2	Eq. (12.2-17)	THETA2	-	-
-	-	DTLMAX	-15	-
-	-	DTVMAX	50	-
-	-	DZIMAX	0.1	-
h_{cond}	-	HCOND	-	-
-	-	SLMIN	0.02	-
-	-	TUPL	-	-
-	-	WFMIN	0.667 x	-
-	-	WFMINS	WFMIN	-
w_{fs}	Eq. (12.5-13)	WFS00	WF0	-
f_m	Eq. (12.7-1)	FRACP	-	-
-	-	FRUPT	-	-
w_{fe}	Eq. (12.5-9)	WF0	-	-
-	-	DTSI	3	-
-	-	DTS	10	-
A_{fr}	Eq. (12.6-9)	AFRV	0.316	5-22
b_{fr}	Eq. (12.6-9)	BFRV	-0.25	5-22
-	-	XMINL	0.05	-
-	-	DTCMIN	-	-
-	-	WFMIND	-	-
-	-	WFMNSD	-	-
$f_{2\phi}$	Eq. (12.6-10)	FVAPM	0	5-7
A_{fr}	-	AFRF	0.316	5-22
b_{fr}	-	BFRF	-0.25	5.7
-	-	TPDMIN	1.0×10^{-5}	5-22
A^*	Eq. (12.7-1)	AGSRLS	-	-
K_g	Eq. (12.7-1)	GASKOR	1.5 - 2.0	-
		PGRMIN	-	-
M_{wg}	Eq. (12.7-31)	GASMW	100 - 130	-
		TMFAIL	-	-

Since the SAS4A/SASSYS-1 calculation can accommodate up to eight different types of fuel material, the number of fuel types used in a given data deck must be specified. This is done through the variable **IFUEL1**. The fuel types must then each be designated as belonging to the core or the blankets. If, in a given channel, the blankets are of one homogeneous material and the core is of another, the quantity **IFUELV** specifies which of the **IFUEL1** fuel types is assigned to the core, while **IFUELB** (Bloc 51, 16) indicates which is assigned to the radial blankets. If, on the other hand, the core and/or blankets each consist of radial bands of different types of fuel, the variable **IFUEL1** is used to match up each radial zone with the appropriate fuel type in both the core and the blankets.

The cladding properties are used only in tabular form by the code. Although only one type of cladding material is used in any one channel, different types may be used in different channels. The total number of types (maximum of three) for a particular run is designated in **ICLAD1**, and the type assigned to each channel is specified in **ICLADV**. The cladding thermal conductivity values are input to **EXKTB** as a function of temperature **EXKTM**. The specific heats are entered in **CPCTAB** with the corresponding temperatures **CPCTEM**. The product of the

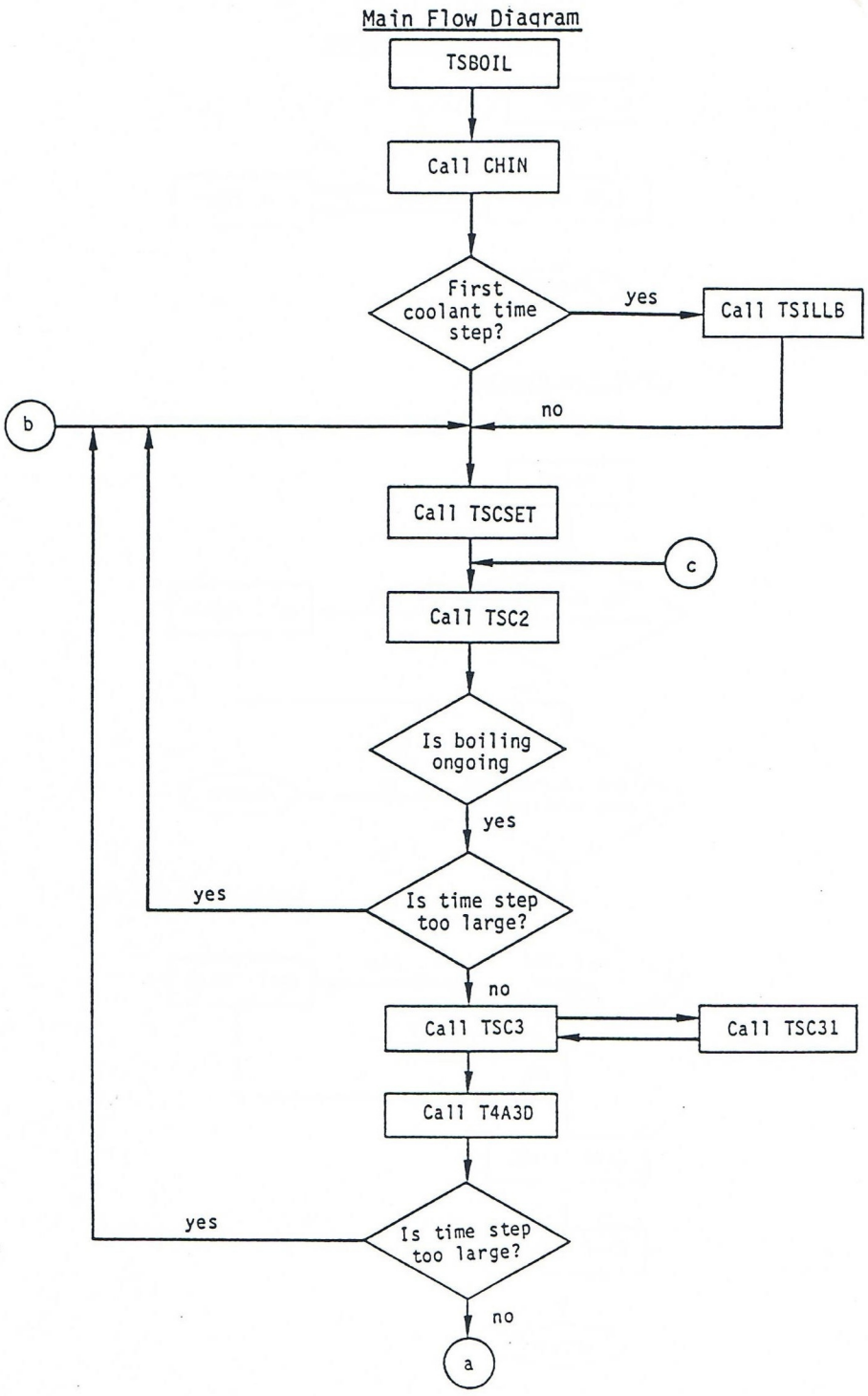


Figure 12.10.1: Flow Chart of the Voiding Model Coding

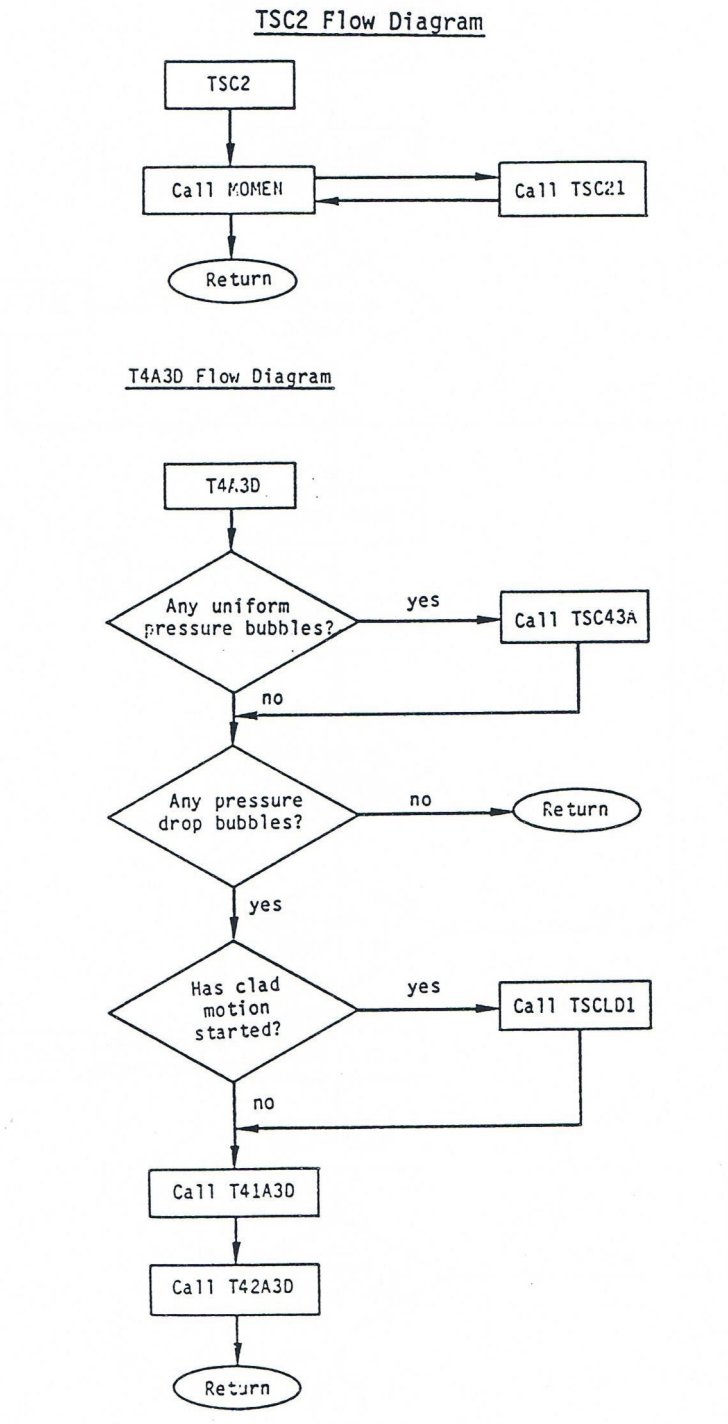


Figure 12.10.2: Flow Chart of the Voiding Model Coding (Cont'd)

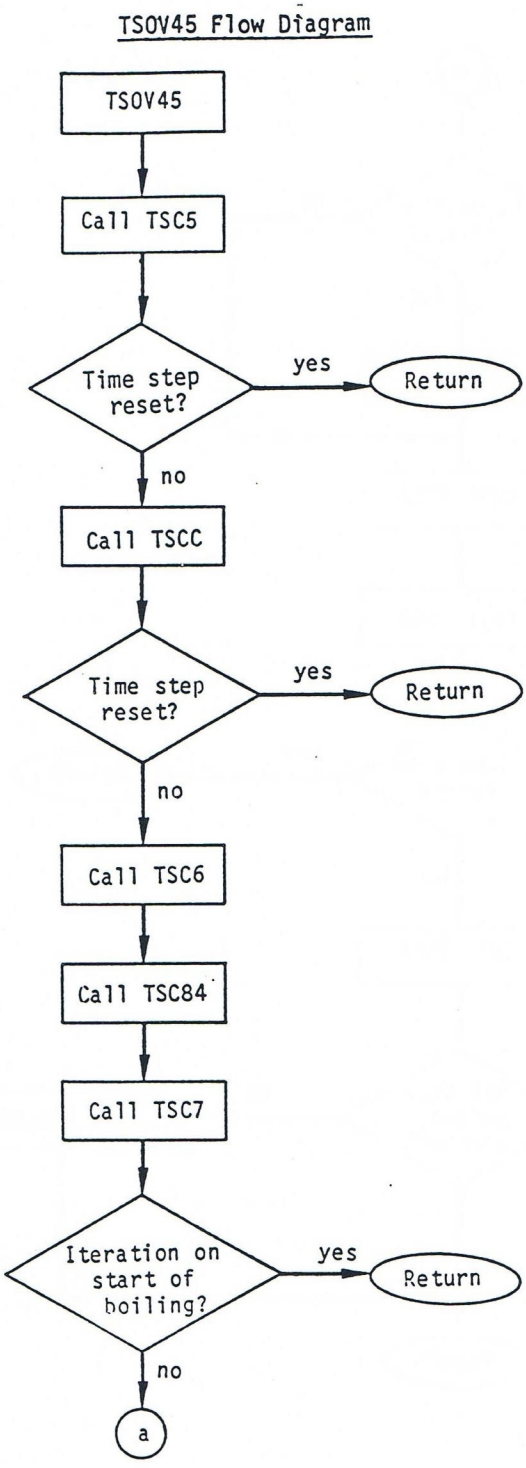


Figure 12.10.3: Flow Chart of the Voiding Model Coding (Cont'd)

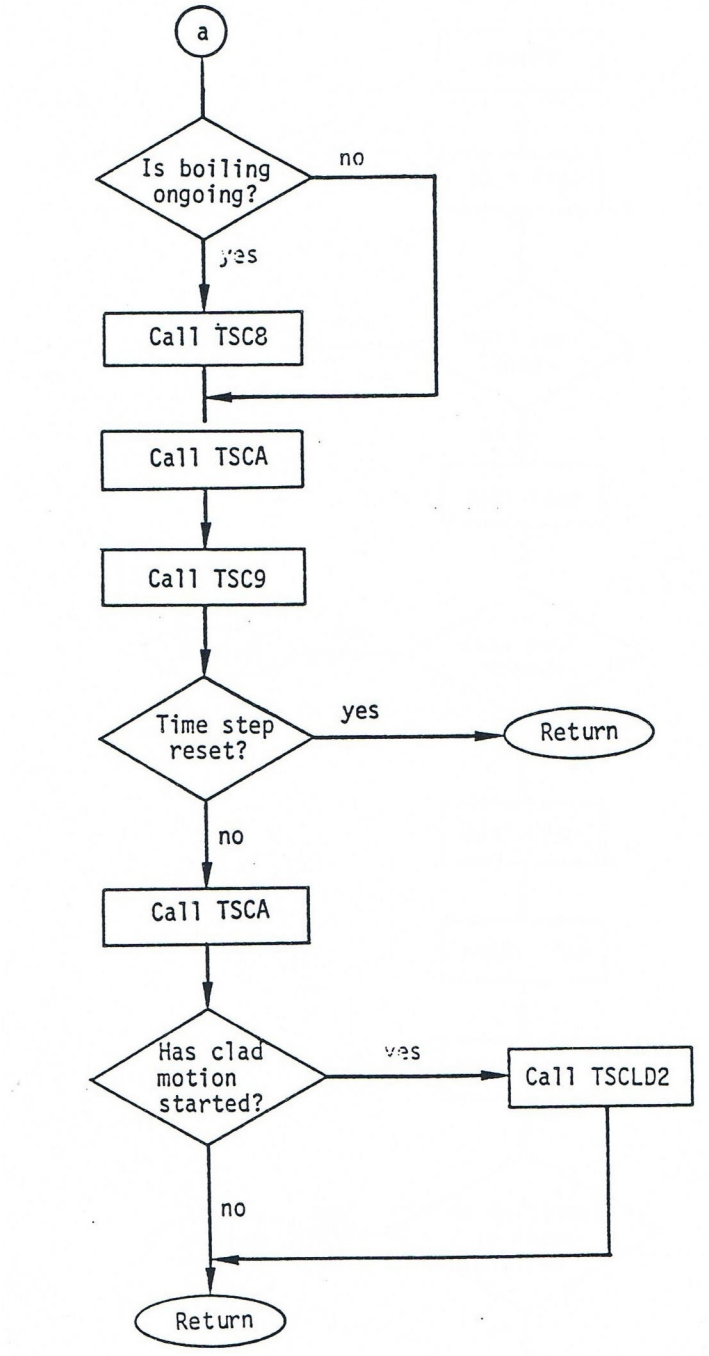


Figure 12.10.4: Flow Chart of the Voiding Model Coding (Cont'd)

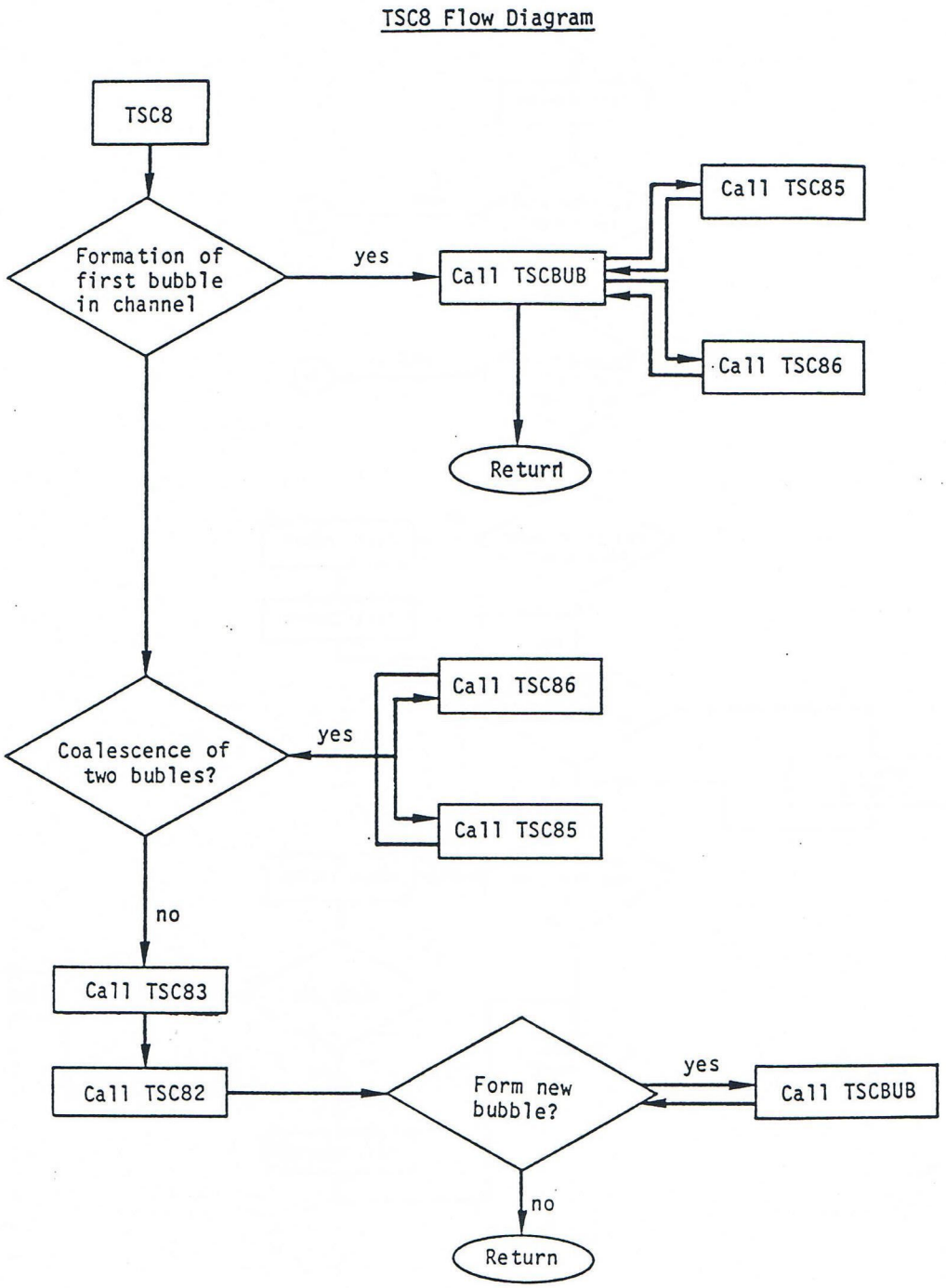


Figure 12.10.5: Flow Chart of the Voiding Model Coding (Cont'd)

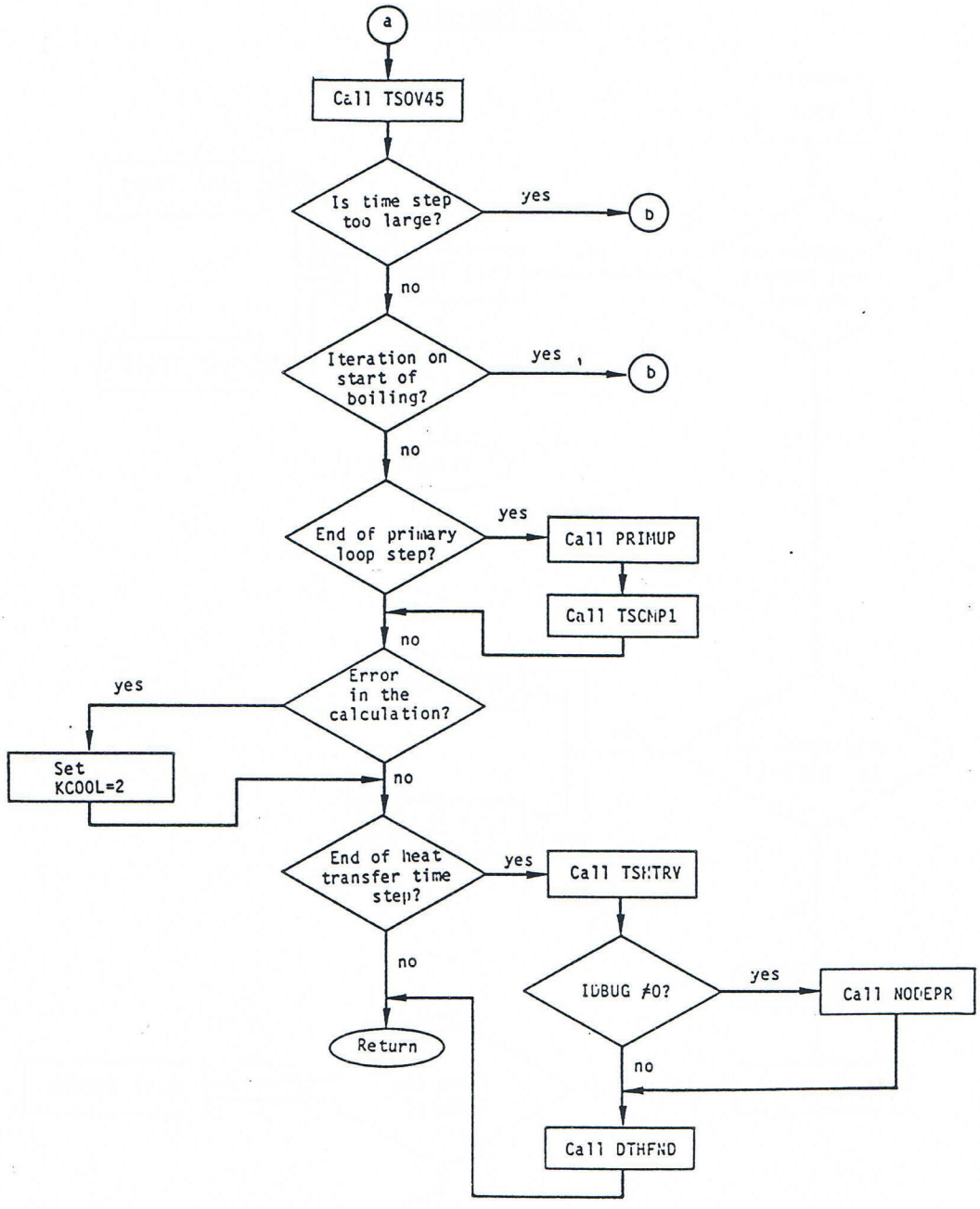


Figure 12.10.6: Flow chart of the Voiding Model Coding

cladding specific heat and the density is entered as a variable in **CROETB** as a function of temperature **CROETM**. In addition, the specific heat of the cladding at the solidus temperature is specified in **CE**. Finally, the density of the solid cladding at an input reference temperature is given for each channel in **DENSS**.

The material properties for the structure and reflector regions are also input in tabular form. In the case of the structure, the code allows different sets of properties to be used for the inner and outer structure nodes. However, the properties are constant for each node within a particular zone and are independent of temperature. The thermal conductivity for the inner structure node is stored in **XKSTIZ**, and that for the outer node is input to **XKSTOZ**. Similarly, the product of the density and the heat capacity for the inner node is read into **RHOCSI**, and for the outer node, this quantity is stored in **RHOCSO**. Of course, the structure properties can vary from channel to channel.

The materials properties for the reflectors are also considered to be constant within a given zone and independent of temperature. However, unlike the case of the structure, the reflector variables are not allowed to differ between the inner and outer segments. The thermal conductivity is given in **XKRF**, and the product of density and heat capacity is input to **RHOCR**. These, too, may be different in different channels.

A couple of properties are input for the gas plenum. The product of density and heat capacity for the plenum gas is found in **RHOCG**, and the thermal resistance of the gas is given in **RG**. These are both independent of temperature and variable form channel to channel.

As indicated earlier in the text, several fluid heat-transfer coefficients are required in order to model the heat transfer to the sodium. For the bond which exists between the fuel and the cladding, the heat-transfer coefficient is given by a correlation for which the coefficients are read into **AHBPAR**, **BHBPAR**, **CHBPAR**, and **HBPAR**. The limits on the bond heat-transfer coefficient are set in **HBMAX** and **HBMIN**. Also, the option exists for calculating the coefficient by the method used in the SAS3D code; this option is invoked by setting the variable **IHGAP** to zero. The convective heat-transfer coefficient of the sodium is given by the Dittus-Boettler correlation, for which the coefficients **C1**, **C2**, and **C3** must be input. If condensation of sodium vapor onto the cladding or structure is occurring, the user-supplied condensation heat-transfer coefficient **HCOND** must be employed.

Much of the input to the voiding model is concerned with the geometry of the problem. Most of this information can be varied from channel to channel, but a few geometry-related features are consistent throughout the system. First, of course, the number of channels must be specified in **NCHAN**. The location of the gas plenum must also be set through the variable **IPLUP**; **IPLUP**=0 fixes the plenum above the core, while **IPLUP**=1 designates that it does below the core. The heights of the lower coolant plenum inlet and the upper coolant plenum outlet (**ZPLENL** and **ZPLENU**) must be input. Finally, because modeling exists in SASSYS-1 to allow material expansion with temperature, the temperature at which the input geometry information is measured must be known; this is specified in **TR**.

Much of the geometric information is initially assumed constant within each axial zone and therefore is read in zone by zone. This requires specification of the number of zones. The pin section always constitutes ne axial zone, and each reflector contains from one to five zones. The number of zones in the lower axial reflector is read into **NREFB**, while that for the upper reflector is given in **NREFT**. The data which are input on a zonal basis are the coolant flow area per fuel pin **ACCZ**, the hydraulic diameter **DHZ**, and the length of each zone **ZONEL**. The structure variables designating the thickness of the inner node (**DSTIZ**) and the outer node (**DSTOZ**) are read in by zone, as is the structure perimeter **SRFSTZ**. Finally, the thicknesses of the outer and inner reflector segments (**DRFO** and **DRFI**, respectively) and the reflector perimeter **SER** are all input by zone.

Some of the remaining geometric information is read in by axial mesh segment. The input information required about the axial mesh structure includes the number of mesh segments in the gas plenum **NPLN**, the number of segments in the upper and lower blankets (**MZUB** and **MZLB**, respectively), and the total number of segments in each axial zone (**NZNODE**). Quantities for which a value must be entered for each axial mesh segment in the pin region include **AXHI**, the length of each axial segment; **RBR**, the inner radius of the cladding; **RER**, the outer radius of the cladding; **RINFP**, the fuel inner radius; and **ROUTFP**, the fuel outer radius.

Some geometric variables define the radial composition of the problem. These include data such as the number

of radial temperature segments in the fuel (NT), the number of concentric rings of different types of fuel in the core and blankets (NZONF), the radial mesh boundaries of the fuel rings (NODSUM), and a flag to specify whether the radial mesh is set up on an equal radius or equal mass basis (IEQMAS).

A few additional pieces of information on the problem geometry must be input. The cladding inner and outer radii in the gas plenum (RBRPL and RERPL) must be specified separately from the values in the pin region (RBR and RER, mentioned above), as must values for the nominal cladding inner and outer radii (RBR0 and RER0). The nominal radii simply represent average values in the pin region; usually the cladding radii in the pin are all equal at the beginning of a run, and the nominal radii are just set to these values. The length of the fission-gas plenum is given in PLENL, the area of the coolant channel plus the pin is set in AREAPC, and the volume fraction of the coolant in the channel is represented in VFC. The inertial terms below and above the subassembly are read into DZIAB and DZIAT, and the orifice coefficients at the entrance and exit of each axial zone are stored in XKORI. The number of pins per subassembly is given in NPIN, and the number of subassemblies represented by the channel is designated in NSUBAS.

Another category of input data is that of temperature initialization. The inlet temperature T0TAB must be specified as a function of time T0TME, with NT0TAB being the number of entries in the inlet temperature table. The temperature of any liquid sodium which reenters the subassembly from the top is fixed at TUPL. The solidus and liquidus temperatures of the fuel are given in TFSOL and TFLIQ, respectively, while those for the cladding are defined in TESOL and TELIQ. The flag which directs whether an Eulerian or a Lagrangian liquid temperature calculation is performed prior to flow reversal is set in ILAG, and the amount of superheat required before boiling can begin is fixed in DTS. The superheat required for formation of new bubbles after voiding has begun is given by DTSI.

Several pieces of information concerning reactor power must be input. The code contains an option for specifying either the power or the external reactivity as a function of time, and the flag IPOWER indicates which quantity is chosen. A total of NPREAT values of the power are read into the array PREATB, with the corresponding times read into array PREATM. The flag IPOWOP determines whether the total steady-state reactor power is computed form the power in the peak axial segment, or vice versa, and the power in the peak axial segment is entered in POW. The fraction of the total reactor power represented by all the SASSYS-1 channels must be given, since some possible channel configurations omit a few low power channels which must be given, since some possible channel configurations omit a few low channels which have a minimal impact on the calculation, and this number is input to FRPR.

The remaining power-related data must be inserted channel by channel. The fraction of the total power due to direct gamma heating must be specified for the structure (GAMSS), coolant (GAMTNC), and cladding (GAMTNE). In addition, the axial and radial shapes of the power distribution are read into PSHAPE and PSHAPR, respectively.

Several conditions concerning the coolant pressure must be designated through the input. The coolant exit pressure is held constant throughout the calculation and is read into variable PX. The magnitude of the coolant inlet pressure at steady state is computed by the code, but, if the PRIMAR-4 option is not invoked, the variation of the normalized inlet pressure with time is either read from a table or calculated from a simple equation. The flag NPRES determines whether the table or the equation will be used. If the equation is chosen, the coefficients PDEC, PDEC1, and PDEC2 must be entered. If the inlet pressure is to be determined from a table, the values are entered in PRETAB, with the corresponding times specified in PRETME. There is also an option to enter the coolant flow rather than the driving pressure in PRETAB. This option is indicated by the flag IFLOW. The user specified coolant flow vs. time is only available if the PRIMAR-1 option is used, IPRION $\neq 4$. With this option, the coolant flow rate in channel IFLOW is specified as a function of time. The outlet plenum pressure is also specified as a function of time. The code back-calculates the inlet plenum pressure required to give the specified flow rate, and this inlet plenum pressure drives the coolant flow in all other channels. Currently, this option does not work after the onset of boiling in channel IFLOW. Finally, the initial gas pressure in the gas plenum at the reference temperature TR is input through P0GAS.

Two other areas for which input data are needed are the expressions for calculation of friction discussed earlier

and the determination of sodium film thickness in voided regions. The coefficients A_{fr} and b_{fr} which are used in the friction expressions are user-input rather than fixed within the code to allow greater flexibility. These include coefficients for the liquid slugs (**AFR** and **BFR**), the sodium vapor (**AFRV** and **BFRV**), and the liquid sodium film (**AFRF** and **BFRF**). Also, the fraction of the two-phase friction factor to be used in voided regions is set in **FVAPM**. The calculation of film thickness requires information about the initial film thickness left on cladding and structure as the voiding front passes by; these numbers are stored in **WF0** for the cladding and **WFS00** for the structure. In addition, the SASSYS-1 model does not try to predict when the sodium films will evaporate completely (zero film thickness); rather, the films are considered to be evaporated when the calculated estimates them to be less than user-input values. In the early part of the calculation, these minimum values are **WFMIN** for the cladding and **WFMINS** for the structure, but after **IFILM** axial segments have experienced dryout, the values in **WFMIND** and **WFMNSD** are used instead.

A few additional pieces fo information are needed inorder to run the calculation. The steady-state coolant mass flow rate is input to **W0** and the fraction of the total reactor flow which is actually represented by core channels is read into **FRFLOW** (again, to accommodate channel maps which do not include all the subassemblies in the reactor; see the discussion of **FRPR** given above). The heat of fusion for the fuel is given in **UFMELT**, and that for the cladding is read into **UEMELT**. Also, the product of the Stefan-Boltzmann constant and the emissivity is assigned to the variable **DEL**.

The options for gas release due to pin failure are determined by **IGASRL** and **IGRLTM**. If IGASRL=0, this option is not used. The method in which pin failure is determined is specified by **IGRLTM**. If **IGRLTM**=0, then DEFORM-5 triggers pin failure in pin group M when the cladding life fraction exceeds **FRUPT**(M). If **IGRLTM**=1, then gas release in pin group M occurs in node **IFAIL** at time **TMFAIL**(M). In addition, the gas release model requires input values for **DZBCGL** and **DZBCGU** for cutting off bubbles after they flow out the top of the subassembly. **GAMGAS** and **GASMW** provide properties of the gas. **FRACP**(M) specifies the fraction of the pins in each group. **AGSRLS** and **GASKOR** determine the gas flow rate through the cladding rip, and **PGRMIN** specifies when gas release will be shut off. The initial plenum gas pressure is determined by **P0GAS** if it is not calculated by DEFORM.

Time-step control involves specification of a number of input variables. The maximum number of main time steps is given in **MAXSTP**, and the maximum problem time is **TIMAX**. The initial main time-step size is **DT0**, and the maximum size after the initiation of voiding is **DTMMXB**. The maximum heat-transfer time-step length after boiling has begun is **DTMXB**. The minimum coolant time-step size before voiding is **DTCMIN**; after the onset of boiling, it is **TPDMIN**.

The different time-step sizes are limited by several input criteria. The main time step must be small enough so that the power does not make a fractional change greater than that specified in **DPWMAX** during the step. The heat-transfer time step must be limited so that, over its span, the fuel temperature does not change by more than **DTFUEL**, the cladding temperature by more than **DTCLAD**, and, prior to voiding, the coolant flow rate by **DWMAX**. The coolant time-step size must be small enough so that the liquid temperature does not change by more than **DTLMAX**, the vapor temperature by more than **DTVMAX**, and the position of any liquid-vapor interface by more than **DZIMAX**.

Printout of results is guided by several input variables. Normally, a standard printout pertaining just to the voiding model is output at the end of designated coolant time steps, and a general standard printout is given at the end of designated main time steps. Boiling printout occurs every **IBLPRN** coolant time steps. General printout is done every **IPO** main time steps up until main time step **IBLPRT**; after step **IBLPRT**, general printout is given every **IPOBOI** time steps. The temperatures in the general printout may be listed in units of degrees Kelvin or degrees centigrade at the discretion of the user through the variable **ITKEL**; however, the user is advised that the labels in the printout will state that temperatures are listed in degrees Kelvin regardless of which option is chosen.

If desired, more detailed output can be obtained through several of the input variables. This is intended primarily for the use of the code developers and outside users who wish to modify the code, but it may on occasion be helpful if problems are encountered in simply running the code. The simplest of this additional output is flagged

by the variable **IHEX**. This number will activate a hexadecimal printout of the sum of the coolant temperatures in a channel. The hexadecimal print is an exact representation of the sum (without truncation) and is therefore useful when comparing two different runs of similar data decks to indicate at what point the two calculations began to diverge. A more extensive output is invoked by the variable **IDBUGV**. Many different printout options are incorporated into **IDBUGV**. The print selected will begin at step **IERSTP**. Prints specialized to the voiding model are also available. The variable **IDBGBL** will trigger the same extensive voiding-related output as the **IDBUGV**=4 option, but it will begin the output on coolant time step **IDBLST** rather than on main time step **IERSTP**. This allows the user to begin a lengthy printout much closer to the time at which the difficulty develops than is possible with **IDBUGV**. The array **IDBFLG** initiates more specialized voiding points. The coolant time step on which each of these prints starts is stored in array **IDBSTP**.

A few additional input variables remain to be discussed. The remaining quantities include **THETA1** and **THETA2**, the coefficients which determine the degree of implicitness of the liquid slug calculation; **SLMIN**, the minimum length of a liquid slug in between two vapor bubbles (minimum slug length is required to stabilize the calculation); and **XMINL**, the bubble length above which the pressure gradient voiding model is used and below which the pressure gradient voiding model is used.

12.12 Sample Output

Samples of the voiding model output are presented in Figure 12.12.1 and Figure 12.12.2. A brief discussion of the data contained in these figures will be given shortly, but first, an explanation of the categories of information listed will be presented. Beginning with the first line of headings in the output, the items listed are as follows:

bubble number
: the row to which this heading belongs gives information about each bubble that is currently in the channel. The bubble number identifies each bubble uniquely and is assigned in the order in which the bubbles are formed.

gas type
: in future modeling, bubbles will be allowed to contain fission gas or fission gas/sodium vapor mixtures, as well as pure sodium vapor, and the composition of the bubble will be identified by the gas type number. For now, all bubbles are gas type 1 (pure sodium vapor).

interface position
: this gives the axial position, in meters, of the lower and upper vapor-liquid interfaces of each bubble in the channel. Bubbles are listed in this section of the output in increasing order of axial position.

velocity
: the velocity with which each interface is moving is given in this column in meters per second.

pressure
: the saturation vapor pressure at each interface in pascals.

vapor and liquid temperatures
: as discussed in the introduction to the chapter, sharp temperature gradients are sometimes present at liquid-vapor interfaces, so at each interface, the code computes a vapor temperature and a liquid temperature a short distance away from the interface. These two temperatures are listed in the output for each interface in kelvins.

boiling time
: This is the time in seconds that has elapsed since the bubble was formed.

cladding film thickness
: this is the thickness in meters of the liquid sodium film which is present on the cladding in

the vapor region by the liquid-vapor interface.

cladding film velocity
in a later version of the code, the liquid film on the cladding will be allowed to move, and this column will show the speed in meters per second with which the film is moving at the interfaces. At the moment, the film velocity is considered to be zero.

structure film thickness
the thickness in meters of the liquid film on the structure in each interfacial vapor region.

axial node
this heading begins a new section of the printout in which information is presented on a node-by-node basis. The axial node number identifies each axial mesh segment interface.

axial position
this is the height along the channel, in meters, of each axial node.

coolant pressure
this is the saturation pressure, in pascals, at each axial node. If the node is contained in a liquid slug, it is the pressure in the liquid, and if the node is part of a vapor bubble, it is the vapor pressure.

coolant flow rate
this is the mass flow rate, in kilograms per second, of the sodium at each node. In liquid nodes, it is the flow rate of the liquid sodium, while in vapor nodes it is that of the sodium vapor.

cladding film velocity
this is the velocity in meters per second of the liquid sodium film on the cladding at each axial node. As mentioned above in the discussion of film velocity at the interfaces, the film velocity is currently set to zero in the code.

cladding film thickness
the thickness in meters of the liquid film on the cladding. This variable is a constraint over a given axial mesh segment (just as the coolant flow area is), and therefore it should actually be designated by axial segment number rather than by axial node number. Since the segment number is just the node number of the node at the bottom of the segment, this variable is listed in the output by the axial node number for segments 1 through MZC-1, with the entry for axial node MZC to be ignored.

clad temperature
this is the cladding temperature, in kelvins, at the midpoint of each axial mesh segment. The comments just made about the listing of the clad film thickness in the output therefore also apply to the cladding temperature; that is, the axial node numbers should be regarded as axial segment numbers for this variable.

saturation temperature
the saturation temperature in kelvins corresponding to the coolant pressure. This variable is calculated at the axial anodes and is listed in that format in the output.

structure temperature
the temperature in kelvins of the structure material. Like the cladding temperature, it is calculated at the mesh segment midpoints and is listed as such in the output.

structure film thickness
the thickness in meters of the liquid sodium film on the structure. It is constant over a given axial mesh segment and is thus listed by segment number.

TCBAR2
: this is the coolant temperature in kelvins at the midpoint of the mesh segment. The heading is jus the name of the variable used in the coding. It is listed by mesh segment, just as the cladding and structure temperatures are.

reflector temperature
: the temperature in kelvins in the reflector regions. It too, is computed at the midpoint of the mesh segment and is given by segment number.

Figure 12.12.1 and Figure 12.12.1 show two examples of data presented in the format just discussed. Both are taken from a run made for the analysis of a loss-of-flow case. Figure 12.12-1 gives information about conditions in the lead channel a short time after voiding initiation. Already, twelve bubbles have formed and the first seven have either collapsed or been swept out of the channel. At this point, all the bubbles are small enough to use the uniform vapor pressure model, as shown by the axially constant pressure and vapor temperature in each bubble. Most of the bubbles are contained in one axial mesh segment, but one, bubble 10, spans a segment interface; this produces the mass flow rate of zero in node 32, since the flow rate is assumed to be zero in the uniform-vapor-pressure model.

Figure 12.12.2 presents data about the same channel at a more advanced stage of voiding. Only one bubble is present in the channel, although more than ninety have been formed since voiding started. The pressure-gradient model is now in operation, and the upper bubble interface is past the top of the channel. A significant region of the cladding has experienced film dryout (i.e., the film thickness calculated by the code has dropped below a user-input minimum value and therefore the thickness has been set to zero), resulting in the high cladding temperatures shown in that region. On the other hand, the structure is cold enough that liquid has condensed on it, increasing the thickness of the film. This has also happened in some areas of the cladding. Note the large temperature difference at the lower void interface between the vapor and liquid temperatures, more than 100 K.

12.13 Physical Properties of Sodium

The expressions used in SAS4A/SASSYS-1 to model the physical properties of sodium as functions of temperature are derived from data correlations generated by Padilla *[12-11]* and by Fink and Leibowitz *[12-12]*. These correlations were developed using the most recent measurements available, namely those of Bhise and Bonilla *[12-13]* and Da Gupta and Bonilla *[12-14]*. The experiments described in Refs. *[12-13]* and *[12-14]* resulted in the first measurements of sodium properties in the temperature range from about 1250 K up to the critical point (2503.3 K). Because of this new information, sodium properties can be modeled in SAS4A/SASSYS-1 over a much wider temperature range than was possible in previous versions of the SAS code.

The correlations given in Refs. *[12-11]* and *[12-12]* could have been used directly in SASSYS-1. However, in order to increase computational efficiency, least-squares approximations to these correlations were generated and incorporated in the code. These approximations introduce little additional error into the calculation (they fit the correlations to 1.5% or better in all cases), and they require less central processing time than do the more complex expressions given in the references. In addition, one polynomial fit is made over the temperature range of interest per sodium property. The references often present two or more correlations (each valid over different sections of the temperature range) for a given property. Implementing different correlations over different temperature ranges for a single property would require computationally expensive branching logic in the code; this is avoided by the polynomial fits.

Because most sections of the code do not require properties at temperatures above about 90% of the critical point (2270 K), the polynomial fits do not extend past 2270 K in order to avoid the difficulty of fitting a polynomial to the rapid changes that occur in most properties near the critical point. On the low end, the temperature range goes down to 590 K, well below any temperature needed in fast reactor accident analyses.

The polynomial expressions for each property are listed below, along with a brief explanation of their origin. In all cases, the temperature T is in Kelvins.

100 STEPS, 0.14657 SECONDS, SINCE BOILING STARTED IN CHANNEL 2 13 BUBBLES FORMED

337 MAIN TIME STEPS HAVE BEEN COMPLETED AT 0.16223D+02 SECONDS, COOLANT TIME STEP IS 0.48092000D-03

BUBBLE NUMBER	GAS TYPE	INTERFACE POSITION	VELOCITY	PRESSURE	TEMPERATURE VAPOR	TEMPERATURE LIQUID	BOILING TIME	* CLAD FILM * THICKNESS	* CLAD FILM * VELOCITY	STRUCTURE FILM THICKNESS	STRUCTURE FILM VELOCITY
12	1	1.7739	2.326	0.1679D+06	1218.053	1222.611	0.00365	1.459D-04	0.0	1.460D-04	0.0
		1.7760	3.265	0.1679D+06	1218.053	1222.589		1.459D-04	0.0	1.460D-04	0.0
11	1	1.7961	3.265	0.1657D+06	1216.416	1222.199	0.01048	1.458D-04	0.0	1.461D-04	0.0
		1.8124	5.994	0.1657D+06	1216.416	1221.867		1.458D-04	0.0	1.461D-04	0.0
10	1	1.8325	5.996	0.1585D+06	1211.038	1220.252	0.02234	1.461D-04	0.0	1.462D-04	0.0
		1.8326	5.926	0.1585D+06	1211.038	1217.704		1.463D-04	0.0	1.461D-04	0.0
9	1	1.9008	5.926	0.1758D+06	1223.768	1215.004	0.02548	1.464D-04	0.0	1.461D-04	0.0
		1.9028	2.900	0.1758D+06	1223.768	1214.304		1.464D-04	0.0	1.461D-04	0.0

AXIAL NODE	AXIAL POSITION	COOLANT PRESSURE	COOLANT TEMPERATURE	COOLANT FLOW RATE	CLAD FILM VELOCITY	CLAD FILM THICKNESS	* * TEMPERATURE * * CLAD	SATURATION	STRUCTURE	STRUCTURE FILM THICKNESS	TCBAR2	REFLECTOR TEMPERATURE
38	2.579	1.543D+05	1141.490	4.741D-02	0.0	1.460D-04	0.0	1207.72	0.0	1.460D-04	570.745	0.0
37	2.459	1.581D+05	1152.257	4.741D-02	0.0	1.460D-04	1141.22	1210.70	1145.10	1.460D-04	1146.873	1141.22
36	2.339	1.619D+05	1161.596	4.741D-02	0.0	1.460D-04	1152.11	1213.62	1156.01	1.460D-04	1156.927	1152.11
35	2.219	1.657D+05	1175.037	4.741D-02	0.0	1.460D-04	1162.95	1216.48	1166.93	1.460D-04	1168.342	1162.95
34	2.099	1.696D+05	1184.933	4.741D-02	0.0	1.460D-04	1173.85	1219.29	1177.89	1.460D-04	1180.010	1173.85
33	1.979	1.734D+05	1201.749	4.741D-02	0.0	1.460D-04	1185.53	1222.05	1189.15	1.460D-04	1193.341	1185.53
32	1.859	1.585D+05	1211.038	0.0	0.0	1.460D-04	1198.70	1211.04	1202.83	1.460D-04	1207.453	1198.70
31	1.696	1.691D+05	1222.722	3.547D-02	0.0	1.460D-04	1223.15	1218.93	1214.50	1.460D-04	1217.806	
30	1.614	1.703D+05	1222.610	3.547D-02	0.0	1.460D-04	1223.06	1219.84	1211.55	1.460D-04	1222.666	
29	1.533	1.716D+05	1220.829	3.547D-02	0.0	1.460D-04	1223.03	1220.74	1210.10	1.460D-04	1221.719	
28	1.451	1.728D+05	1217.075	3.547D-02	0.0	1.460D-04	1221.42	1221.65	1207.34	1.460D-04	1218.952	
27	1.399	1.737D+05	1201.438	3.547D-02	0.0	1.460D-04	1222.95	1222.23	1197.71	1.460D-04	1209.256	
26	1.347	1.745D+05	1183.053	3.547D-02	0.0	1.460D-04	1208.07	1222.82	1180.97	1.460D-04	1192.245	
25	1.294	1.753D+05	1161.938	3.547D-02	0.0	1.460D-04	1190.36	1223.40	1161.52	1.460D-04	1172.495	
24	1.189	1.770D+05	1112.190	3.547D-02	0.0	1.460D-04	1157.57	1224.57	1126.61	1.460D-04	1137.064	
23	1.084	1.786D+05	1054.158	3.547D-02	0.0	1.460D-04	1106.19	1225.75	1073.49	1.460D-04	1083.174	
22	0.977	1.803D+05	990.528	3.547D-02	0.0	1.460D-04	1046.64	1226.92	1013.54	1.460D-04	1022.343	
21	0.871	1.820D+05	924.342	3.547D-02	0.0	1.460D-04	982.01	1228.08	949.61	1.460D-04	957.435	
20	0.766	1.837D+05	858.866	3.547D-02	0.0	1.460D-04	915.30	1229.23	884.88	1.460D-04	891.604	
19	0.662	1.853D+05	797.433	3.547D-02	0.0	1.460D-04	849.88	1230.36	822.61	1.460D-04	828.150	
18	0.558	1.870D+05	742.927	3.547D-02	0.0	1.460D-04	789.07	1231.47	765.90	1.460D-04	770.180	
17	0.506	1.878D+05	718.971	3.547D-02	0.0	1.460D-04	747.32	1232.02	727.69	1.460D-04	730.949	
16	0.455	1.886D+05	697.375	3.547D-02	0.0	1.460D-04	722.79	1232.57	705.74	1.460D-04	708.173	
15	0.404	1.894D+05	678.166	3.547D-02	0.0	1.460D-04	700.63	1233.12	686.32	1.460D-04	687.770	
14	0.323	1.907D+05	670.010	3.547D-02	0.0	1.460D-04	677.57	1233.96	673.31	1.460D-04	674.088	
13	0.161	1.932D+05	659.944	3.547D-02	0.0	1.460D-04	667.13	1235.63	664.50	1.460D-04	664.977	
12	0.0	1.958D+05	654.150	3.547D-02	0.0	1.460D-04	658.27	1237.28	656.89	1.460D-04	657.047	
11	-0.086	1.969D+05	654.150	3.547D-02	0.0	1.460D-04	654.15	1238.00	654.15	1.460D-04	654.150	
10	-0.173	1.980D+05	654.150	3.547D-02	0.0	1.460D-04	654.15	1238.72	654.15	1.460D-04	654.150	
9	-0.259	1.991D+05	654.150	3.547D-02	0.0	1.460D-04	654.15	1239.44	654.15	1.460D-04	654.150	
8	-0.346	2.002D+05	654.150	3.547D-02	0.0	1.460D-04	654.15	1240.15	654.15	1.460D-04	654.150	
7	-0.432	2.013D+05	654.150	3.547D-02	0.0	1.460D-04	654.15	1240.86	654.15	1.460D-04	654.150	
6	-0.519	2.024D+05	654.150	3.547D-02	0.0	1.460D-04	654.15	1241.56	654.15	1.460D-04	654.150	
5	-0.637	2.040D+05	654.150	3.547D-02	0.0	1.460D-04	654.15	1242.52	654.15	1.460D-04	654.150	654.15
4	-0.755	2.055D+05	654.150	3.547D-02	0.0	1.460D-04	654.15	1243.47	654.15	1.460D-04	654.150	654.15
3	-0.873	2.070D+05	654.150	3.547D-02	0.0	1.460D-04	654.15	1244.42	654.15	1.460D-04	654.150	654.15
2	-0.991	2.085D+05	654.150	3.547D-02	0.0	1.460D-04	654.15	1245.36	654.15	1.460D-04	654.150	654.15
1	-1.109	2.218D+05	654.150	3.547D-02	0.0	1.460D-04	654.15	1253.37	654.15	1.460D-04	654.150	654.15

Figure 12.12.1: Example of a Printout from the Voiding Model Shortly after Voiding Initiation

CHANNEL 2 SAS4A 0.0 TRANSIENT PIN CALCULATION FOR HAC 9/10/83 12.04.36 PAGE1383
JUNE,1982

1730 STEPS, 1.82099 SECONDS, SINCE BOILING STARTED IN CHANNEL 2 89 BUBBLES FORMED

500 MAIN TIME STEPS HAVE BEEN COMPLETED AT 0.17893D+02 SECONDS, COOLANT TIME STEP IS 0.32528500D-02

BUBBLE NUMBER	GAS TYPE	INTERFACE POSITION	VELOCITY	PRESSURE	TEMPERATURE VAPOR	TEMPERATURE LIQUID	BOILING TIME	* CLAD FILM * THICKNESS	* CLAD FILM * VELOCITY	STRUCTURE FILM THICKNESS	STRUCTURE FILM VELOCITY
87	1	0.3828	0.359	0.2059D+06	1243.733	1150.336	0.88208	3.710D-04	0.0	4.315D-04	0.0
		2.6749	57.632	0.1633D+06	1215.066	1205.610		1.460D-04	0.0	1.460D-04	0.0

AXIAL NODE	AXIAL POSITION	COOLANT PRESSURE	COOLANT TEMPERATURE	COOLANT FLOW RATE	CLAD FILM VELOCITY	CLAD FILM THICKNESS	* * TEMPERATURE * * CLAD	SATURATION	STRUCTURE	STRUCTURE FILM THICKNESS	TCBAR2	REFLECTOR TEMPERATURE
38	2.585	1.641D+05	1215.247	8.589D-04	0.0	1.460D-04	0.0	1215.25	0.0	1.460D-04	1215.188	0.0
37	2.465	1.646D+05	1215.613	1.037D-03	0.0	2.920D-04	1210.83	1215.61	1205.65	2.087D-04	1215.430	1210.83
36	2.345	1.655D+05	1216.303	1.147D-03	0.0	2.900D-04	1213.62	1216.30	1198.77	2.441D-04	1215.958	1213.62
35	2.225	1.667D+05	1217.151	1.252D-03	0.0	2.849D-04	1214.57	1217.15	1202.29	2.270D-04	1216.727	1214.57
34	2.105	1.680D+05	1218.172	1.346D-03	0.0	2.569D-04	1215.92	1218.17	1206.60	2.019D-04	1217.661	1215.92
33	1.985	1.697D+05	1219.371	1.425D-03	0.0	2.332D-04	1217.47	1219.37	1211.17	1.783D-04	1218.772	1217.47
32	1.865	1.716D+05	1220.726	1.491D-03	0.0	2.093D-04	1219.09	1220.73	1215.54	1.593D-04	1220.049	1219.09
31	1.702	1.773D+05	1224.847	1.449D-03	0.0	1.367D-04	1223.26	1224.85	1222.04	1.379D-04	1222.786	
30	1.620	1.803D+05	1226.924	1.403D-03	0.0	1.222D-04	1226.83	1226.92	1226.51	1.402D-04	1225.885	
29	1.539	1.834D+05	1229.022	1.329D-03	0.0	1.070D-04	1229.40	1229.02	1229.38	1.420D-04	1227.973	
28	1.457	1.865D+05	1231.120	1.216D-03	0.0	8.643D-05	1232.07	1231.12	1232.48	1.403D-04	1230.071	
27	1.405	1.876D+05	1231.872	1.216D-03	0.0	0.0	1355.97	1231.87	1232.49	1.641D-04	1231.496	
26	1.352	1.887D+05	1232.617	1.216D-03	0.0	0.0	1391.92	1232.62	1230.04	2.015D-04	1232.244	
25	1.300	1.898D+05	1233.355	1.216D-03	0.0	0.0	1425.02	1233.35	1226.38	2.448D-04	1232.986	
24	1.194	1.920D+05	1234.808	1.216D-03	0.0	0.0	1464.02	1234.81	1220.33	2.920D-04	1234.082	
23	1.088	1.941D+05	1236.233	1.216D-03	0.0	0.0	1492.51	1236.23	1207.06	2.920D-04	1235.520	
22	0.981	1.963D+05	1237.625	1.217D-03	0.0	0.0	1462.63	1237.63	1186.89	2.920D-04	1236.929	
21	0.874	1.984D+05	1238.973	1.217D-03	0.0	0.0	1408.65	1238.97	1164.37	2.920D-04	1238.299	
20	0.769	2.004D+05	1240.275	1.217D-03	0.0	0.0	1324.64	1240.28	1136.64	2.920D-04	1239.624	
19	0.663	2.060D+05	1243.834	4.170D-04	0.0	3.806D-05	1250.62	1243.83	1104.17	2.920D-04	1242.055	
18	0.559	2.067D+05	1244.238	-1.389D-04	0.0	1.446D-04	1252.03	1244.24	1068.86	2.920D-04	1244.036	
17	0.507	2.064D+05	1244.049	-3.122D-04	0.0	2.266D-04	1250.22	1244.05	1032.08	2.920D-04	1244.143	
16	0.455	2.061D+05	1243.845	-4.080D-04	0.0	2.735D-04	1247.81	1243.84	1015.05	2.920D-04	1243.947	
15	0.404	2.059D+05	1243.752	-4.165D-04	0.0	2.904D-04	1244.23	1243.75	986.64	2.920D-04	1243.798	
14	0.323	2.062D+05	894.669	4.610D-03	0.0	1.460D-04	1116.97	1243.94	955.46	1.460D-04	943.268	
13	0.161	2.072D+05	713.862	4.610D-03	0.0	1.460D-04	808.45	1244.55	743.25	1.460D-04	804.265	
12	0.0	2.082D+05	674.318	4.610D-03	0.0	1.460D-04	695.05	1245.17	680.60	1.460D-04	694.090	
11	-0.086	2.088D+05	651.927	4.610D-03	0.0	1.460D-04	661.96	1245.51	661.88	1.460D-04	663.123	
10	-0.173	2.093D+05	663.838	4.610D-03	0.0	1.460D-04	659.52	1245.85	653.53	1.460D-04	657.883	
9	-0.259	2.099D+05	659.469	4.610D-03	0.0	1.460D-04	661.16	1246.19	658.58	1.460D-04	661.654	
8	-0.346	2.104D+05	659.146	4.610D-03	0.0	1.460D-04	660.20	1246.53	654.75	1.460D-04	659.307	
7	-0.432	2.109D+05	661.291	4.610D-03	0.0	1.460D-04	660.57	1246.86	657.31	1.460D-04	660.218	
6	-0.519	2.115D+05	660.171	4.610D-03	0.0	1.460D-04	660.72	1247.20	655.67	1.460D-04	660.731	
5	-0.637	2.123D+05	649.506	4.610D-03	0.0	1.460D-04	655.61	1247.66	655.01	1.460D-04	654.839	655.61
4	-0.755	2.130D+05	659.296	4.610D-03	0.0	1.460D-04	653.66	1248.12	653.85	1.460D-04	654.401	653.66
3	-0.873	2.138D+05	647.692	4.610D-03	0.0	1.460D-04	654.32	1248.57	654.28	1.460D-04	653.494	654.32
2	-0.991	2.145D+05	662.482	4.610D-03	0.0	1.460D-04	654.12	1249.03	654.10	1.460D-04	655.087	654.12
1	-1.109	2.155D+05	653.821	4.610D-03	0.0	1.460D-04	657.14	1249.60	655.31	1.460D-04	658.152	657.13

Figure 12.12.2: Example of a Printout from the Voiding Model at an Advanced Stage of Boiling

Heat of Vaporization (λ in J/kg)

$$\lambda = A_1 + A_2 T + A_3 T^2 + A_4 T^3 \tag{12.13-1}$$

$A_1 = 5.3139 \times 10^6$

$A_2 = -2.0296 \times 10^3$

$A_3 = 1.0625$

$A_4 = -3.3163 \times 10^{-4}$

The expression for λ is a fit to two equations recommended by Padilla; the first (valid below 1644 K) is from Golden Tokar's *[12-15]* quasichemical approach, while the second (for T > 1644 K) is the model of Miller, Cohen, and Dickerman *[12-16]*, as adjusted by Padilla. The polynomial fit approximates both equations to within 1%.

Saturation Vapor Pressure (P_s in Pa)

$$\ln(P_s) = A_5 - \frac{A_6}{T} - \frac{A_7}{T^2} \tag{12.13-2}$$

$A_5 = 2.169 \times 10^1$

$A_6 = 1.14846 \times 10^4$

$A_7 = 3.41769 \times 10^5$

Fink and Leibowitz present the equation

$$\ln(P_s) = 3.0345 \times 10^1 - \frac{13113}{T} - 1.09481 \ln(T) + 1.9777 \times 10^{-4} T \tag{12.13-3}$$

However, SAS4A/SASSYS-1 requires an expression for temperature as a function of pressure, which must be obtained by inverting the equation, giving pressure as a function of temperature. Since this formula cannot be inverted directly, the least-squares fit shown above was used. The polynomial approximates the reference equation to within 1.2%.

Saturation Temperature (T_s in K)

$$T_s = \frac{A_8}{A_9 + \sqrt{A_{10} + A_{11} \ln(P_s)}} \tag{12.13-4}$$

$A_8 = 2A_7$

$A_9 = -A_6$

$A_{10} = A_6^2 + 4A_5A_7$

$A_{11} = -4A_7$

$3.5 \leq P_s \leq 1.6 \times 10^7$ Pa

Liquid Density (ρ_l in kg/m^3)

$$\rho_l = A_{12} + A_{13}T + A_{14}T^2 \tag{12.13-5}$$

$A_{12} = 1.00423 \times 10^3$

$A_{13} = -0.21390$

$A_{14} = -1.1046 \times 10^{-5}$

This equation fits two empirical equations recommended by Fink and Leibowitz to within 0.5%. Below 1644 K, they suggest the equation of Stone, et al. *[12-17]*, while above 1644 K, they use a model of their own.

Vapor Density (ρ_v in kg/m^3)

$$\rho_v = P_s(\frac{A_{15}}{T} + A_{16} + A_{17}T + A_{18}T^2 + A_{19}T^3 + A_{20}T^4) \tag{12.13-6}$$

$A_{15} = 4.1444 \times 10^{-3}$

$A_{16} = -7.4461 \times 10^{-6}$

$A_{17} = 1.3768 \times 10^{-8}$

$A_{18} = -1.0834 \times 10^{-11}$

$A_{19} = 3.8903 \times 10^{-15}$

$A_{20} = -4.922 \times 10^{-19}$

with P_s in pascals. This equation substitutes for the two correlations presented by Padilla, one generated using the quasichemical approach (below 1644 K), the other (above 1644 K) being the Clapeyron equation. The polynomial is accurate to 1.5%.

Liquid Heat Capacity (C_l in J/kg-K)

$$C_l = \frac{A_{28}}{(T_c - T)^2} + \frac{A_{29}}{T_c - T} + A_{30} + A_{31}(T_c - T) + A_{32}(T_c - T)^2 \tag{12.13-7}$$

$A_{28} = 7.3898 \times 10^5$

$A_{29} = 3.1514 \times 10^5$

$A_{30} = 1.1340 \times 10^3$

$A_{31} = -2.2153 \times 10^{-1}$

$A_{32} = 1.1156 \times 10^{-4}$

$T_c = 2503.3$ K = the critical temperature

This equation fits to within 1.5% the data generated by Padilla using a thermodynamic relation from Rowlinson *[12-18]*.

Vapor Heat Capacity (C_g in J/kg-K)

$$C_g = A_{33} + A_{34}T + A_{35}T^2 + A_{36}T^3 + A_{37}T^4 + A_{38}T^5 + A_{39}T^6 \tag{12.13-8}$$

$A_{33} = 2.1409 \times 10^3$

$A_{34} = -2.2401 \times 10^1$

$A_{35} = 7.9787 \times 10^{-2}$

$A_{36} = -1.0618 \times 10^{-4}$

$A_{37} = 6.7874 \times 10^{-8}$

$A_{38} = -2.1127 \times 10^{-11}$

$A_{39} = 2.5834 \times 10^{-15}$

Padilla recommends a quasichemical approach below 1644 K and a relation of Rowlinson's above that point. This polynomial approximates both correlations to better than 1%.

Liquid Adiabatic Compressibility (β_s in Pa^{-1})

$$\beta_s = A_{40} + \frac{A_{41}}{T_c - T} \tag{12.13-9}$$

$A_{40} = -5.4415 \times 10^{-11}$

$A_{41} = 4.7663 \times 10^{-7}$

Padilla computed values for this quantity from experimental measurements of sonic velocity below 1773 K and from an empirical correlation due to Grosse *[12-19]* above that temperature. The polynomial fits these data to better than 0.1%.

Liquid Thermal Expansion Coefficient (α_p in K^{-1})

$$\alpha_p = A_{42} + \frac{A_{43}}{T_c - T} + \frac{A_{44}}{(T_c - T)^2} + \frac{A_{45}}{(T_c - T)^3} + \frac{A_{46}}{(T_c - T)^4} + \frac{A_{47}}{(T_c - T)^5} \tag{12.13-10}$$

$A_{42} = 2.5156 \times 10^{-6}$

$A_{43} = 0.79919$

$A_{44} = -6.9716 \times 10^2$

$A_{45} = 3.3140 \times 10^5$

$A_{46} = -7.0502 \times 10^7$

$A_{47} = 5.4920 \times 10^9$

Fink and Leibowitz express α_p by applying a thermodynamic relation to the data of Das Gupta and Bonilla. The above expression fits their correlation to better than 0.7%.

Liquid Thermal Conductivity (k_l in W/m-K)

$$k_l = A_{48} + A_{49}T + A_{50}T^2 + A_{51}T^3 \tag{12.13-11}$$

$A_{48} = 1.1045 \times 10^2$

$A_{49} = -6.5112 \times 10^{-2}$

$A_{50} = 1.5430 \times 10^{-5}$

$A_{51} = -2.4617 \times 10^{-9}$

The expression approximates the data given by Fink and Leibowitz to within 0.5%. This information comes from experimental data below 1500 K and extrapolated values above 1500 K generated by a method described by Grosse *[12-20]*.

Liquid Viscosity (μ_l in Pa-s)

$$\mu_l = A_{52} + \frac{A_{53}}{T} + \frac{A_{54}}{T^2} + \frac{A_{55}}{T^3} \tag{12.13-12}$$

$A_{52} = 3.6522 \times 10^{-5}$

$A_{53} = 0.16626$

$A_{54} = -4.56877 \times 10^1$

$A_{55} = 2.8733 \times 10^4$

Fink and Leibowitz present this variable as a combination of experimental data below 1200 K and extrapolated values based on a method suggested by Grosse *[12-21]*. Their results are approximated by the above polynomial to within 0.5%.

Enthalpy of Saturated Liquid (H in J/kg)

$$H = A_{56} + A_{57}T + A_{58}T^2 + A_{59}T^3 \tag{12.13-13}$$

$A_{56} = -111136.04$

$A_{57} = 1722.2578$

$A_{58} = -0.45544483$

$A_{59} = 1.4692883 \times 10^{-4}$

The enthalpy of saturated sodium is only used in the fuel failure models *PLUTO* and *LEVITATE*.

12.14 References

12-1. F. E. Dunn, et al., Unpublished information, Argonne National Laboratory, 1974.

12-2. F. E. Dunn, et al., Unpublished information, Argonne National Laboratory, 1975.

12-3. J. E. Cahalan and D. R. Ferguson, Eds., Unpublished information, Argonne National Laboratory, 1977.

12-4. M. A. Grolmes and H. K. Fauske, "Modeling of Sodium Expulsion with FREON-II", *Am. Soc. Mech. Eng. Fluids Engineering, Heat Transfer and Fabrication Conf.*, Detroit, Michigan, Paper 70-HT-24, 1970.

12-5. S. W. Eisenhawer and R. W. Albrecht, "An Investigation of Forced Convection Boiling of Sodium at Low Heat Fluxes in Electrically heated Seven Pin Bundles", *Int. Mtg. Fast Reactor Safety and Related Physics*, Chicago, Illinois, CONF-761001, 1976.

12-6. G. Klein and F. E. Dunn, "SAS3D Analysis of Natural Convection Boiling Behavior in the Sodium Boiling Test Facility", *Trans. Am. Nucl. Soc.*, vol. 83, p. 517, 1979.

12-7. A. W. Cronenberg, H. K. Fauske, D. G. Bankoff, and D. T. Eggen, "A Single Bubble Model for Sodium Expulsion from a Heated Channel", *Nucl. Eng. Design*, vol. 16, p. 285, 1971.

12-8. A. W. Cronenberg, "A Thermohydrodynamic Model for Molten UO_2-Na Interaction, Pertaining to Fast-reactor Fuel-failure Accidents", ANL-7947, Argonne National Laboratory, Argonne, Illinois, 1972.

12-9. E. G. Schlechtendahl, "Theoretical Investigations on Sodium Boiling in Fast Reactors", *Nucl. Sci. Eng.*, vol. 41, p. 99, 1970.

12-10. G. B. Wallis, *One-dimensional Two-phase Flow*, McGraw-Hill, 1969.

12-11. A. Padilla, "High Temperature Thermodynamic Properties of Sodium", HEDL-TME 77-27, Hanford Engineering Development Laboratory, Richland, WA, February 1978.

12-12. J. K. Fink and L. Lebowitz, "Thermophysical Properties of Sodium", ANL-CEN-RSD-79-1, Argonne National Laboratory, Argonne, Illinois, May 1979.

12-13. V. S. Bhise and C. F. Bonilla, "The Experimental Vapor Pressure and Critical Point of Sodium", *Proc. Int. Conf. on Liquid Metal Technology in Energy Production*, CONF 760503-P2, May 1976.

12-14. S. Das Gupta and C. F. Bonilla, *Experimental High-Temperature Coefficients of Compressibility and Expansivity of Liquid Sodium and Other Related Properties*, Columbia, University, 1977.

12-15. G. H. Golden and J. V. Tokar, "Thermophysical Properties of Sodium", ANL-7323, Argonne National Laboratory, Argonne, Illinois, August 1967.

12-16. D. Miller, A. B. Cohen, and C. E. Dickerman, "Estimation of Vapor and Liquid Density and Heat of Vaporization of the Alkali Metals to the Critical Point", *Proc. Int. Conf. on the Safety of Fast Breeder Reactors*, September 1967.

12-17. J. P. Stone, et al., "High-temperature Properties of Sodium", NRL-6241, Naval Research Laboratory, 1965.

12-18. J. S. Rowlinson, *Liquids and Liquid Mixtures*, 2nd Ed., Plenum Press, New York, 1969.

12-19. A. V. Grosse, "Simple Empirical Relationship Between the Compressibility of Ideal Liquids and Temperature up to the Critical Point", *Nature Physical Science*, vol. 232, p. 170, 1971.

12-20. A. V. Grosse, *Rev. Hautes Temp. Refract.*, vol. 3, pp. 115-146, 1966.

12-21. A. V. Grosse, *Science*, vol. 14, p. 1438, 1965.

12-22. M. M. El-Wakil, *Nuclear Heat Transport*, International Textbook Company, Scranton, PA, 1971.

12-23. W. M. Rohsenow and H. Y. Choi, *Heat, Mass, and Momentum Transfer*, Prentice-Hall, Inc., Englewood Cliffs, NJ, 1981.

12.15 Nomenclature

Symbol	Definition	Units
$A_c\,(z,t)$	Coolant flow area	m^3
A_{fr}	User-input quantity in the friction factor expression	–
AA_0	Coefficient in the liquid slug momentum equation	Pa
b_{fr}	User-input quantity in the friction factor expression	–
BB_0	Coefficient in the liquid slug momentum equation	$m^{-1} - s^{-1}$
C_l	Specific heat of liquid sodium	J/kg-K
$D_h\,(z,t)$	Hydraulic diameter	m
E_i	Heat flow to the vapor through the liquid-vapor Interfaces	J
E_t	Total heat energy added to a vapor bubble in a time step	J
E_{es}	Heat flow to the vapor from the cladding and structure	J
$F_c\,(z,t)$	Condensation momentum loss term	Pa/m
$f_v\,(z,t)$	Friction factor	–
$F_{ce}\,(z,t)$	Condensation momentum loss term to cladding film	Pa/m
$F_{cs}\,(z,t)$	Condensation momentum loss term to structure film	Pa/m
f_{rwb}, f_{rwt}	Liquid film rewetting factors	–
g	Gravitational acceleration	m/s^2
$h_{ec}\,(z)$	Heat-transfer coefficient from the cladding surface to the sodium vapor	W/m^2K
$h_{sc}\,(z)$	Heat-transfer coefficient from the structure surface to the sodium vapor	W/m^2K
h_{cond}	User-supplied condensation heat-transfer coefficient	W/m^2K
I_1, I_2, I_3, I_4, I_5	Coefficients in the liquid slug momentum equation	–
K	Bubble number	–
K_{vn}	Number of bubbles	–
$k_e\,(z)$	Cladding thermal conductivity	W/m-K
$k_l\,(z)$	Thermal conductivity of liquid sodium at Temperature T(z)	W/m-K
$k_{or}\,(z,t)$	Orifice pressure drop term	–
$p\,(z,t)$	Vapor pressure	Pa
$P_e\,(z)$	Perimeter of cladding - $2\pi r_e$	m
$Q\,(z,t)$	Total heat flux per unit volume $= Q_e + Q_s$	W/m^3
$\widetilde{Q}\,(t)$	Total heat flow rate into the sodium vapor	W
$q\,(z,t)$	Surface heat flow per unit area	W/m^2
$Q_c\,(c)$	Volume source due to direct heating by neutrons and gamma rays	W/m^3
$Q_e\,(z,t)$	Heat flow rate per unit volume through the cladding	W/m^3
$q_e\,(z,t)$	Cladding-to-vapor heat flux	W/m^2
$Q_s\,(z,t)$	Heat flow rate per unit volume through the structure	W/m^3
$q_s\,(z,t)$	Structure-to-vapor heat flux	W/m^2
$Q_{es}\,(t)$	Heat flow rate from cladding and structure	W
$r_e\,(z)$	Nominal radius of cladding	m
$R_{ec}\,(z,t)$	Thermal resistance between the cladding and the coolant	m^2K/W
$R_{sc}\,(z,t)$	Thermal resistance between the structure and the coolant	m^2K/W
s	Vapor heat transfer surface area	m^2
$T\,(z,t)$	Coolant temperature	m
t	Time	m
$T\,(z,t)$	Cladding temperature	m

continues on next page

Table 12.15.1 – continued from previous page

Symbol	Definition	Units
$T_l\,(t)$	Liquid sodium temperature at liquid-vapor interface	m
$T_s\,(z,t)$	Structure temperature	m
$T_v\,(t)$	Sodium vapor temperature at liquid-vapor interface	m
$v\,(z,t)$	Velocity	m/s
v_{fe}	Velocity of liquid film on the cladding (currently fixed at zero)	m/s
v_{fs}	Velocity of liquid film on the structure (currently fixed at zero)	m/s
$W\,(z,t)$	Mass flow rate	kg/s
$w_{fe}\,(z,t)$	Thickness of liquid film on the cladding	m
$w_{fs}\,(z,t)$	Thickness of liquid film on the structure	m
z	Axial height	m
$z_i\,(1,t,K)$	Height of lower liquid-vapor interface in bubble K	m
$z_i\,(2,t,K)$	Height of upper liquid-vapor interface in bubble K	m
α	Thermal diffusivity of liquid sodium $= k_l\rho_l C_l$	m^2s
$\gamma\,(z,t)$	Ratio of cladding surface area to coolant volume	m-1
$\gamma_2\,(z,t)$	Ratio of surface area of structure to surface area of cladding	–
ΔI_j	Coefficients in the liquid slug momentum equation	–
Δr_e	Nominal cladding thickness	m
Δt	Time step	s
θ_1, θ_2	Coefficients which set the degree of implicitness of the liquid slug calculation	–
$\lambda\,(z,t)$	Heat of vaporization	J/kg
$\mu\,(z,t)$	Sodium vapor viscosity	$Pa-s$
$\rho_t\,(z)$	Density of liquid sodium	kg/m^3
$\rho_v\,(z)$	Density of sodium vapor	kg/m^3
Subscripts and Superscripts		
b	Bottom	
c	Coolant	
e	Cladding	
I	Interface quantity	
J	Global node number of the node under consideration	
$J1$	Global node number of the lower interface of the bubble under consideration	
$J2$	Global node number of the upper interface of the bubble under consideration	
l	Liquid	
L	Lower interface	
s	Structure	
t	Top	
u	Upper interface	
v	Vapor	
$1, 2$	Beginning and end of current time step	

12.16 Appendices

12.16.1 Derivation of the Expression for the Spatial Derivative of the Liquid Temperature at a Liquid-Vapor Interface

The derivation of Eq. (12.5-44), the expression for $\frac{\partial T_l}{\partial \xi}\big|_{\xi=0}$, from the general heat conduction equation involves a somewhat lengthy process which will be described in this appendix. The starting point is Eq. (12.5-39), the general heat conduction equation, written as

$$\frac{\partial^2 T\left(\xi,t'\right)}{\partial \xi^2}+\frac{Q\left(\xi,t'\right)}{k_l}=\frac{1}{\alpha}\frac{\partial T\left(\xi,t'\right)}{\partial t'} \tag{12.16-1}$$

where the nomenclature is defined in Section 12.5. As discussed in Section 12.5, the boundary conditions for Eq. (12.16-1) are $T\left(0,t'\right)$ equal to the vapor temperature and $T\left(\xi,t'\right)$ finite, while the initial condition is that $T\left(\infty,0\right)$ is known. If the Laplace transform is applied to Eq. (12.16-1), the resulting equation is

$$\frac{\partial^2 \overline{T}\left(\xi,s\right)}{\partial \xi^2}+\frac{\overline{Q}\left(\xi,s\right)}{k_l}=\frac{1}{\alpha}\left(s\overline{T}\left(\xi,s\right)-T\left(\xi,0\right)\right) \tag{12.16-2}$$

where $\overline{T}$ and $\overline{Q}$ and are the Laplace transforms of T and Q, respectively, and s is the transform variable. Eq. (12.16-2) will be somewhat easier to work with if the function $g\left(\xi,s\right)$ is defined as

$$g\left(\xi,s\right)=-\frac{\overline{Q}\left(\xi,s\right)}{k_l}-\frac{T\left(\xi,0\right)}{\alpha} \tag{12.16-3}$$

so that the transformed equation becomes

$$\frac{\partial^2 \overline{T}}{\partial \xi^2}-\frac{s}{\alpha}\overline{T}=g \tag{12.16-4}$$

The arguments ξ and s have been suppressed in Eq. (12.16-4) for simplicity of notation. Eq. (12.16-4) is a second-order partial differential equation in ξ, which can be solved in a straightforward manner. First, one additional notational simplification will be introduced by defining the variable q as

$$q=\sqrt{\frac{s}{\alpha}} \tag{12.16-5}$$

which gives

$$\frac{\partial^2 \overline{T}}{\partial \xi^2}-q^2\overline{T}=g \tag{12.16-6}$$

The solution to this equation can be divided into a homogeneous solution $\overline{T}_h$ and a particular solution $\overline{T}_p$

$$\overline{T}=\overline{T}_h+\overline{T}_p \tag{12.16-7}$$

The homogeneous solution is just the solution to the equation

$$\frac{\partial^2 \overline{T}_h}{\partial \xi}-q^2\overline{T}_h=0 \tag{12.16-8}$$

so that $\overline{T}_h$ has the form

$$\overline{T}_h=C_1\exp\left(q\xi\right)-C_2\exp\left(-q\xi\right) \tag{12.16-9}$$

where C_1 and C_2 are constants to be determined from the boundary conditions.

Finding the functional form of the particular solution is a more complex process than was finding the form of the homogeneous solution. The particular solution satisfies the equation

$$\frac{\partial^2 \overline{T}_p}{\partial \xi^2} - q^2 \overline{T}_p = g \tag{12.16-10}$$

which can be rewritten as

$$\left(\frac{\partial}{\partial \xi} - q\right)\left(\frac{\partial}{\partial \xi} + q\right)\overline{T}_p = g \tag{12.16-11}$$

If the function u is defined as

$$u = \frac{\partial \overline{T}_p}{\partial \xi} + q\overline{T}_p \tag{12.16-12}$$

then Eq. (12.16-11) takes the simpler form

$$\frac{\partial u}{\partial \xi} - qu = g \tag{12.16-13}$$

Eq. (12.16-13) is a simple first-order equation which is easily solved. First, multiply the equation by $e^{-q\xi}$:

$$\frac{\partial u}{\partial \xi} e^{-q\xi} - que^{-q\xi} = ge^{-q\xi} \tag{12.16-14}$$

which can be written more simply as

$$\frac{\partial}{\partial \xi}\left(ue^{-q\xi}\right) = ge^{-q\xi} \tag{12.16-15}$$

Eq. (12.16-15) can then be solved for u by integrating from 0 to ξ,

$$ue^{-q\xi} = \int_0^\xi ge^{-q\xi'} d\xi' \tag{12.16-16}$$

or

$$u = e^{q\xi} \int_0^\xi ge^{-q\xi'} d\xi' \tag{12.16-17}$$

If this expression is substituted into Eq. (12.16-12), the following equation for the particular solution results:

$$\frac{\partial \overline{T}_p}{\partial \xi} + q\overline{T}_p = e^{q\xi} \int_0^\xi ge^{-q\xi'} \partial \xi' \tag{12.16-18}$$

Eq. (12.16-18) is solved using the same procedure as was employed to solve Eq. (12.16-18) for u, only this time, the multiplier is $e^{q\xi}$, giving

$$\frac{\partial}{\partial \xi}\left(\overline{T}_p e^{q\xi}\right) = e^{2q\xi} \int_0^\xi ge^{-q\xi''} d\xi'' \tag{12.16-19}$$

which gives the following expression for $\overline{T}_p$:

$$\overline{T}_p = e^{-q\xi} \int_0^\xi e^{2q\xi'} \left(\int_0^{\xi'} ge^{-q\xi'} d\xi''\right) d\xi'' \tag{12.16-20}$$

Using integration by parts, Eq. (12.16-20) can be modified from a double integral to the difference of two integrals, giving

$$\overline{T}_p = e^{q\xi}\int_0^{\xi}\frac{g}{2q}e^{-q\xi'}\,d\xi' - e^{-q\xi}\int_0^{\xi}\frac{g}{2q}e^{q\xi'}\,d\xi' \tag{12.16-21}$$

The Laplace transform of the liquid temperature is then just the sum of the expression for $\overline{T}_n$ from Eq. (12.16-9) and that for $\overline{T}_p$ from Eq. (12.16-21):

$$\begin{aligned}\overline{T} &= \overline{T}_h + \overline{T}_p \\ &= C_1\exp(q\xi) + C_2\exp(-q\xi) + \exp(q\xi)\int_0^{\xi}\frac{g}{2q}\exp(-q\xi')\,d\xi' \\ &\quad - \exp(-q\xi)\int_0^{\xi}\frac{g}{2q}\exp(q\xi')\,d\xi'\end{aligned} \tag{12.16-22}$$

The constants C_1 and C_2 in Eq. (12.16-22) can be evaluated by imposing the boundary conditions stated at the beginning of the appendix. The condition at $\xi = 0$ gives

$$\overline{T}(0,s) = C_1 + C_2 \tag{12.16-23}$$

while that for $\xi \to \infty$ requires that

$$C_1 + \int_0^{\infty}\frac{g}{2q}\exp(-q\xi')\,d\xi' = 0 \tag{12.16-24}$$

If these are applied to Eq. (12.16-22), the resulting expression for $\overline{T}$ is

$$\begin{aligned}\overline{T}(\xi,s) &= -\exp(q\xi)\int_0^{\infty}\frac{g}{2q}\exp(-q\xi')\,d\xi' + \overline{T}(0,s)\exp(-q\xi) \\ &+\exp(-q\xi)\int_0^{\infty}\frac{g}{2q}\exp(-q\xi')\,d\xi' + \exp(q\xi)\int_0^{\xi}\frac{g}{2q}\exp(-q\xi')\,d\xi' \\ &\quad -\exp(-q\xi)\int_0^{\xi}\frac{g}{2q}\exp(q\xi')\,d\xi'\end{aligned} \tag{12.16-25}$$

Taking the derivative of Eq. (12.16-25) with respect to ξ and evaluating the result at $\xi = 0$ gives

$$\begin{aligned}\left.\frac{\partial\overline{T}}{\partial\xi}\right|_{\xi=0} &= -q\int_0^{\infty}\frac{g}{2q}\exp(-q\xi')\,d\xi' - q\overline{T}(0,s) \\ &\quad -q\int_0^{\infty}\frac{g}{2q}\exp(-q\xi')\,d\xi' + \frac{g}{2q} - \frac{g}{2q}\end{aligned} \tag{12.16-26}$$

which, substituting in the functions represented by q and g, produces

$$\begin{aligned}\left.\frac{\partial\overline{T}}{\partial\xi}\right|_{\xi=0} &= -\overline{T}(0,s)\sqrt{\frac{s}{\alpha}} + \int_0^{\infty}\frac{\overline{Q}(\xi',s)}{k_l}\exp\left(-\xi'\sqrt{\frac{s}{\alpha}}\right)d\xi' \\ &\quad + \int_0^{\infty}\frac{T(\xi',0)}{\alpha}\exp\left(-\xi'\sqrt{\frac{s}{\alpha}}\right)d\xi'\end{aligned} \tag{12.16-27}$$

All that remains now is to take the inverse Laplace transform of Eq. (12.16-27). Representing the inverse transform by L^{-1}, the necessary inverse transforms are

$$L^{-1}\left[\left.\frac{\partial\overline{T}}{\partial\xi}\right|_{\xi=0}\right] = \left.\frac{\partial T}{\partial\xi}\right|_{\xi=0} \tag{12.16-28}$$

$$
\begin{aligned}
L^{-1}\left[\overline{T}\left(0,s\right)\sqrt{\frac{s}{\alpha}}\right] &= L^{-1}\left[\overline{T}\left(0,s\right)\frac{s}{\sqrt{s\alpha}}\right] \\
&= L^{-1}\left[s\overline{T}\left(0,s\right)\frac{1}{\sqrt{\alpha}\sqrt{s}}\right] \\
&= \int_0^{t'}\frac{1}{\sqrt{\pi\alpha}}\frac{1}{\sqrt{t'-\tau}}\left(\frac{\partial T(0,\tau)}{\partial\tau}\right)d\tau
\end{aligned}
\tag{12.16-29}
$$

by convolution

$$
\begin{aligned}
L^{-1}\left[T\left(\xi,0\right)\exp\left(-\xi\sqrt{\frac{s}{\alpha}}\right)\right] &= T\left(\xi,0\right)L^{-1}\left[\exp\left(-\frac{\xi}{\sqrt{\alpha}}\sqrt{s}\right)\right] \\
&= T\left(\xi,0\right)\frac{\xi}{2\sqrt{\pi\alpha\left(t'\right)^3}}\exp\left(\frac{-\xi^2}{4\alpha t'}\right)
\end{aligned}
\tag{12.16-30}
$$

$$
L^{-1}\left[\overline{Q}\left(\xi,s\right)\exp\left(-\xi\sqrt{\frac{s}{\alpha}}\right)\right] = \int_0^{t'}Q\left(\xi,\tau\right)\frac{\xi}{2\sqrt{\pi\alpha\left(t'-\tau\right)^3}}\exp\left(-\frac{\xi^2}{4\alpha\left(t'-\tau\right)}\right)d\tau \tag{12.16-31}
$$

The above inverse transforms can be found in any good handbook which lists inverse Laplace transforms. Taking the inverse transform of Eq. (12.16-27) and using the above expressions for the individual inverse transforms finally produces Eq. (12.5-44):

$$
\begin{aligned}
\left.\frac{\partial T\left(t'\right)}{\partial\xi}\right|_{\xi=0} &= -\frac{1}{\sqrt{\pi\alpha}}\int_0^{t'}\frac{\partial T\left(0,\tau\right)}{\partial\tau}\frac{1}{\sqrt{t'-\tau}}d\tau \\
-\frac{T\left(0,0\right)}{\sqrt{\alpha\pi t'}} + \int_0^{\infty}d\xi\int_0^{t'}&\frac{\xi Q\left(\xi,\tau\right)\exp\left[-\frac{\xi^2}{4\alpha\left(t'-\tau\right)}\right]}{2k_l\sqrt{\pi\alpha\left(t'-\tau\right)^3}}d\tau \\
&+\int_0^{\infty}\frac{T\left(\xi,0\right)\xi\exp\left[-\frac{\xi^2}{4\alpha t'}\right]}{2\alpha\sqrt{\pi\alpha\left(t'\right)^3}}d\xi
\end{aligned}
\tag{12.16-32}
$$

12.16.2 Gaussian Elimination Solution of Linearized Vapor-Pressure-Gradient Equations

The series of equations represented by Eq. (12.6-168) is solved as shown in the steps below. Recall that Eq. (12.6-165) $a_{2,J2-1} = a_{2,J1} = 1$.

Forward Reduction

1. Divide the second equation by $b_{2,J1}$:

$$b_{i,J1} \to b_{i,J1}/b_{2,J1}, i = 1\ldots4 \quad g_{J1} \to g_{J1}/b_{2,J1}.$$

2. Divide the third equation by $c_{2,J1}$:

$$c_{i,J1} \to c_{i,J1}/c_{2,J1}, i = 1\ldots4 \quad h_{J1} \to h_{J1}/c_{2,J1}$$

3. Subtract the first equation from the second:

$$b_{i,J1} \rightarrow b_{i,J1} - a_{1,J1} \quad b_{2,J1} \rightarrow 0 \quad g_{J1} \rightarrow g_{J1} - d_{J1}$$

4. Subtract the first equation from the third:

$$c_{1,J1} \rightarrow c_{1,J1} - a_{1,J1} \quad c_{2,J1} \rightarrow 0 \quad h_{J1} \rightarrow h_{J1} - d_{J1}$$

5. Divide the second equation by $b_{1,J1}$:

$$b_{i,J1} \rightarrow b_{i,J1}/b_{1,J1}, i = 1 \ldots 4 \quad g_{J1} \rightarrow g_{J1}/b_{1,J1}$$

6. Divide the third equation by $c_{1,J1}$:

$$c_{i,J1} \rightarrow c_{i,J1}/c_{1,J1}, i = 1 \ldots 4 \quad h_{J1} \rightarrow h_{J1}/c_{1,J1}$$

7. Subtract the second equation from the third:

$$c_{1,J1} \rightarrow 0 \quad c_{3,J1} \rightarrow c_{3,J1} - \widetilde{b}_{3,J1} \quad c_{4,J1} \rightarrow c_{4,J1} - \widetilde{b}_{4,J1}$$
$$h_{J1} \rightarrow h_{J1} - \widetilde{g}_{J1}$$

Repeat Steps 8-15 for $j = J1 + 1, J1 + 2, \ldots, J2 - 1$. Then proceed to Step 16.

8. Divide by $c_{3,j-1}$:

$$c_{4,j-1} \rightarrow c_{4,j-1}/c_{3,j-1} \quad c_{3,j-1} \rightarrow 1 \quad h_{j-1} \rightarrow h_{j-1}/c_{3,j-1} \quad (c_{1,j-1} = 0 \quad c_{2,j-1} = 0)$$

9. Divide by $b_{1,j}$:

$$b_{i,j} \rightarrow b_{i,j}/b_{1,j}, i = 1 \ldots 4 \quad g_j \rightarrow g_j/b_{1,j}$$

10. Divide by $c_{1,j}$:

$$c_{i,j} \rightarrow c_{i,j}/c_{1,j}, i = 1 \ldots 4 \quad h_j \rightarrow h_j/c_{1,j}$$

11. Subtract step 8 from step 9:

$$b_{2,j} \rightarrow b_{2,j} - c_{4,j-1} \quad b_{1,j} \rightarrow 0 \quad g_j \rightarrow g_j - h_{j-1}$$

12. Subtract step 8 from step 10:

$$c_{2,j} \rightarrow c_{2,j} - c_{4,j-1} \quad c_{1,j} \rightarrow 0 \quad h_j \rightarrow h_j - h_{j-1}$$

13. Divide by $b_{2,j}$:

$$b_{i,j} \rightarrow b_{i,j}/b_{2,j}, i = 3, 4 \quad b_{2,j} \rightarrow 1 \quad g_j \rightarrow g_j/b_{2,j}$$

14. Divide by $c_{2,j}$:

$$c_{i,j} \rightarrow c_{i,j}/c_{2,j}, i = 3, 4 \quad c_{2,j} \rightarrow 1 \quad h_j \rightarrow h_j/c_{2,j}$$

15. Subtract step 13 from step 14:

$$c_{i,j} \rightarrow c_{i,j} - b_{i,j}, i = 3, 4 \quad c_{2,j} \rightarrow 0 \quad h_j \rightarrow h_j - g_j$$

Continue

16. Divide by $c_{4,J2-1}$:

$$c_{3,J2-1} \rightarrow c_{3,J2-1}/c_{4,J2-1} \quad c_{4,J2-1} \rightarrow 1 \quad h_{J2-1} \rightarrow h_{J2-1}/c_{4,J2-1}$$

17. Subtract:

$$a_{1,J2-1} \rightarrow a_{1,J2-1} - c_{3,J2-1} \quad a_{2,J2-1} \rightarrow 0 \quad d_{J2-1} \rightarrow d_{J2-1} - h_{J2-1}$$

Back Substitution

18. Solve for upper interface pressure:

$$\Delta p^i(J2) = d_{J2-1}/a_{1,J2-1}$$

19. Solve for upper interface flow:

$$\Delta W^i(J2) = h_{J2-1} - c_{3,J2-1}\Delta p^i(J2)$$

20.

$$\Delta W(J2-1) = \widetilde{g}_{J2-1} - \widetilde{b}_{3,J2-1}\Delta p^i(J2) - \widetilde{b}_{4,J2-1}\Delta W^i(J2)$$

21.

$$\Delta p(J2-1) = h_{J2-1} - c_{4,J2-1}\Delta W(J2-1)$$

Do steps 22 and 23 for $j = J2-2, J2-3, \ldots, J1+1$.

22.

$$\Delta W(j) = g_j - b_{3,j}\Delta p(j+1) - b_{4,j}\Delta W(j+1)$$

23.

$$\Delta p(j) = h_j - c_{4,j}\Delta W(j)$$

24. Solve for lower interface pressure:

$$\Delta p^i(J1) = \widetilde{g}_{J1} - \widetilde{b}_{3,J1}\Delta p(J1+1) - \widetilde{b}_{4,J1}\Delta W(J1+1)$$

25. Solve for lower interface flow:

$$\Delta W^i(J1) = d_{J1} - a_{1,J1}\Delta p^i(J1)$$

CHAPTER

THIRTEEN

CLAP: CLADDING MOTION MODEL

13.1 Introduction and Overview

13.1.1 Background and Description of the CLAP Physical Model

The CLAP Model [13-1, 13-2] computes cladding relocation and phase changes for accident situations in which the fuel geometry is still essentially intact. Such a situation typically arises in undercooling accidents during which coolant voiding and pin dryout occur, followed by rapid heating and subsequent melting of the cladding. In the usual accident scenario, the cladding motion proceeds as follows: (i) the molten cladding may initially rise due to the pressure gradient and viscous shear forces generated by the sodium coolant vapor flow in the dried-out section of the core, (ii) the rising molten cladding then freezes upon encountering cooler structure in the region immediately above the active fuel (either the upper reflector or upper blanket region, depending upon the reactor design), and (iii) the upper frozen-cladding blockage throttles the vapor flow, thus allowing the remaining molten cladding to drain under the influence of gravity. The cladding motion and phase changes significantly affect the subsequent course of events in the scenario both through the influences of cladding-motion-related reactivity changes and frozen-cladding blockages that possibly inhibit later fuel motions.

The available SAS3D cladding relocation model, CLAZAS [13-3], is inflexible in calculation of the consequences for such diverse mechanistic and/or postulated phenomena. This model suffers from several shortcomings in the estimation of cladding accelerations. First, the use of large cladding segments can lead to a significant change in results following relatively minor input changes. Second, the model-dependent Lagrangian mesh leads to difficulties in integrating consistently with the voiding dynamics and fuel motion models. Third, the formalism does not provide flexibility with respect to the mode of cladding refreezing or with respect to variations in the time-dependent coupling to the sodium-vapor dynamics. If treated correctly, the influence of sodium-vapor dynamics could result in a possible display of oscillatory effects.

The CLAP model uses and Eulerian numerical formulation coupling the time-dependent continuity, momentum, and energy equations for a film of moving cladding, the continuity and energy for sodium vapor, and the SAS4A pin heat-transfer calculation. This is accomplished by interconnecting an implicit solution for the low-density sodium vapor with an explicit, modified upwind differencing procedure for the high-density cladding. Axial resolution depends on the mesh spacing selected by the user. The key phenomena influencing cladding motion in CLAP are the degree of interfacial sodium vapor friction and the cladding heat transfer. Input that controls these phenomena permits a variety of experimental situations to be simulated.

As shown in the schematic in Figure 13.1.1, the CLAP model divides the core axially into two types of zones. In the central zone, the cladding is molten and is in motion. In that region, heat is transferred directly from the fuel surface to the molten cladding. In the zones at either end of the fueled region is intact cladding. Due to cladding motion, the intact cladding may be coated with refrozen cladding and/or molten cladding.

13.1.2 CLAP Module Structure and Interaction with Other SAS4A Models

The initiation of CLAP and the decision for adding axial segments to the molten-cladding zone occurs in the fuel-pin model TSHTRV. During cladding motion, the fuel-pin model TSHTRV continues to evaluate the fuel temperatures and, outside of the molten zone, the temperatures of the intact cladding. Coupling between TSHTRV and CLAP allows for heat transfer between the fuel and molten cladding in the molten-cladding zone and between the initial cladding and refrozen (or molten) cladding outside the molten zone.

The CLAP subroutines are called every coolant time step and, as shown in Figure 13.1.2, are called from various locations in the sodium voiding module. The functions of the CLAP subroutines are as follows:

TSCLD1 - (i) initiates CLAP variables, (ii) adds segments to molten zone, (iii) calculates local heat transfer between molten steel and fuel within the molten zone, (iv) calculates local heat transfer between the initial cladding, refrozen cladding (if any), and molten cladding (if any) outside of the molten zone, and (v) computes mass and areas of the refrozen and molten cladding allowing for convection and phase change;

TSCLD2 - solves the momentum equation to obtain the velocity of the molten cladding film and contains the CLAP output and debug print statements;

SODFRC - computes the two-phase friction factor used to evaluate the interfacial shear stress between the sodium vapor and molten steel;

DENSIT - evaluates the cladding melt fraction, temperature, and density as a function of cladding specific internal energy.

The CLAP subroutine TSCLD1 computes the current (local) vapor flow areas and hydraulic diameters, which are used by the sodium voiding model. Also, TSCLD1 may reduce the current coolant time step, if necessary (based on a Courant condition criterion), and require the coolant dynamics model to reevaluate current flow parameters.

The CLAP subroutine TSCLD2 is called later and utilizes updated sodium vapor velocities and pressures to compute the molten cladding velocity for the beginning of the next time step. Local two-phase friction factors (to evaluate the gas/liquid interfacial stress), which are computed by TSCLD2 for the next time step, are passed to the sodium voiding model.

Coupling of CLAP with the reactivity model consists of current cladding mass distributions (axial) computed in TSCLD1 being utilized in the subroutine FEEDBK to evaluate the reactivity change due to cladding relocation.

13.2 Mathematical Model

13.2.1 Sodium Vapor Flow Model

The sodium vapor flow generates the axial pressure gradients and interfacial shear forces, which influence the motion of molten cladding. The cladding motion in turn changes flow areas, hydraulic diameters, and surface roughness which all affect the sodium vapor flow. The governing equations for the sodium vapor, while presented elsewhere (Section 12.6.1) in more detail, are reproduced here to show the intimate coupling between the sodium vapor dynamics and cladding motion. The conservation equations for the sodium vapor are

$$\frac{\partial}{\partial \mathrm{t}}\left(A_{\mathrm{v}}\rho_{\mathrm{v}}\right)+\frac{\partial \mathrm{w}}{\partial \mathrm{z}}=\dot{m}_{\mathrm{v}} \tag{13.2-1}$$

$$\frac{\partial \mathrm{w}}{\partial \mathrm{t}}+\frac{\partial}{\partial \mathrm{z}}\left(\frac{w^2}{A_{\mathrm{v}}}\rho_{\mathrm{v}}\right)+A_{\mathrm{v}}\left[\frac{\partial \mathrm{p}}{\partial \mathrm{z}}+F_{\mathrm{v}}\right]=\begin{cases}\dot{m}_{\mathrm{v}}\dfrac{w}{A_{\mathrm{v}}}\rho_{\mathrm{v}}, & \text{if condensing} \quad \text{(13.2-2a)}\\ 0, & \text{if vaporizing} \quad \text{(13.2-2b)}\end{cases} \tag{13.2-2}$$

where

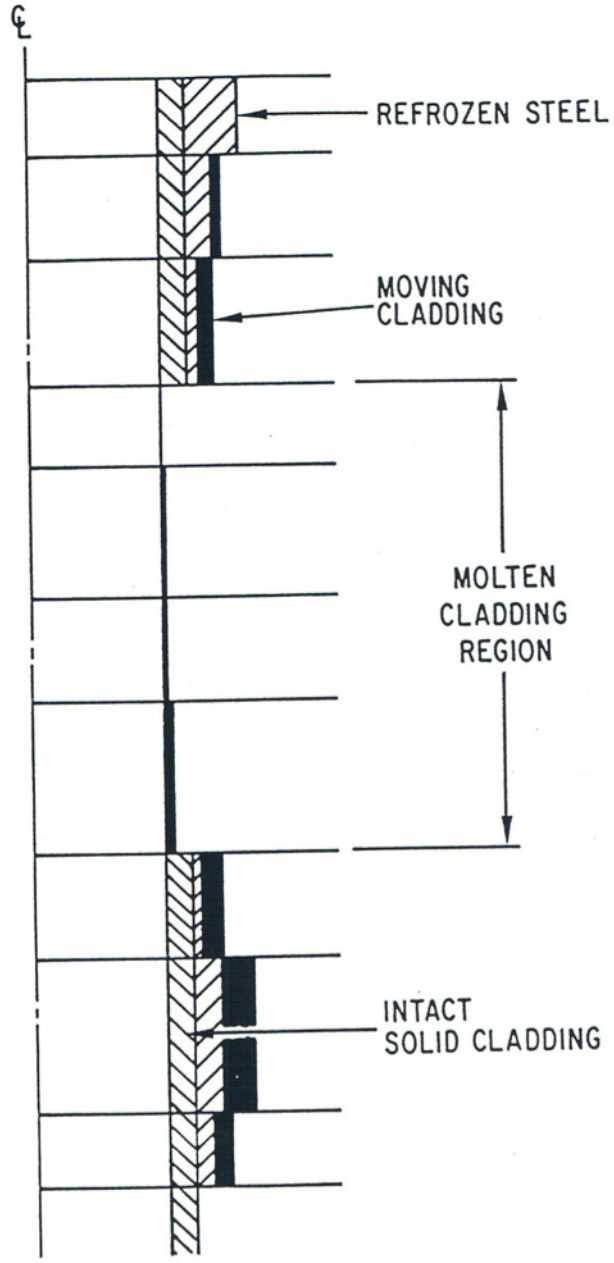


Figure 13.1.1: Schematic of CLAP Geometry Treating Cladding Relocation

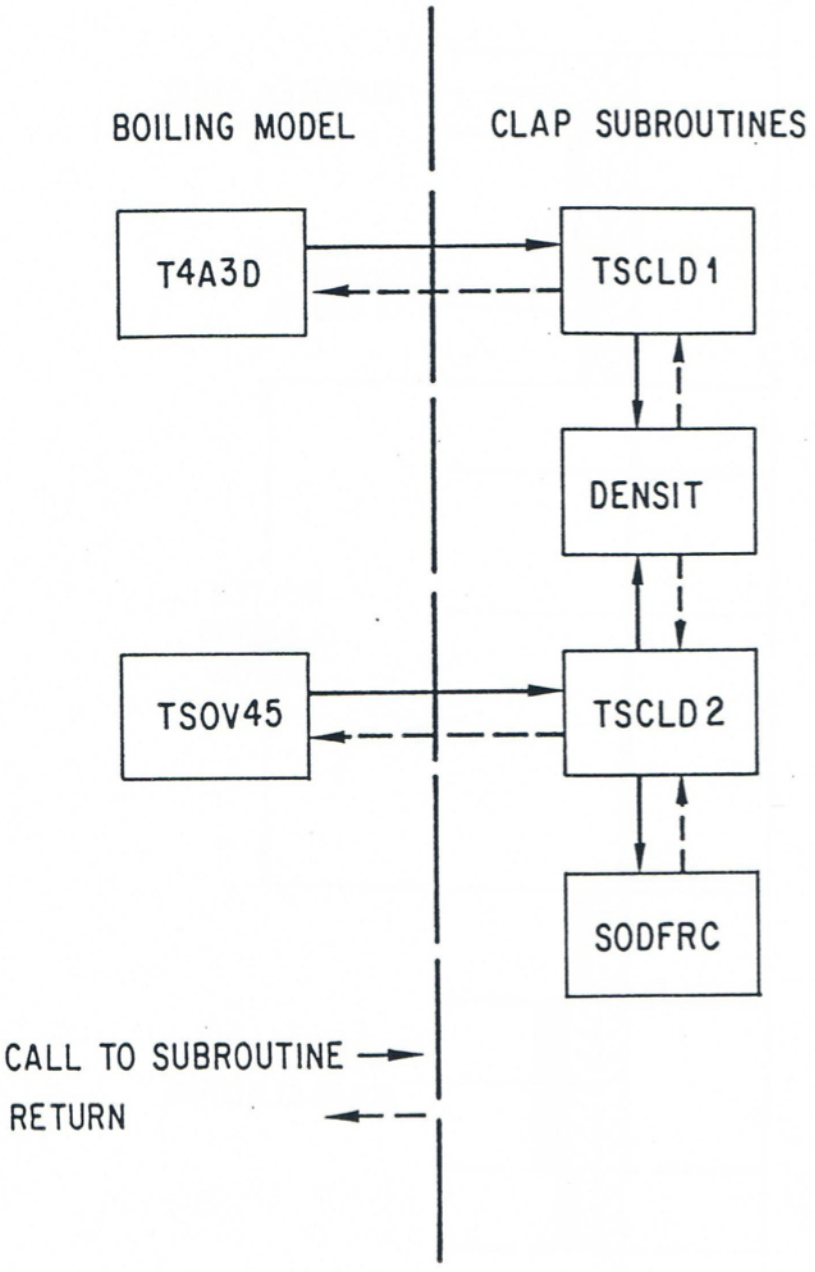


Figure 13.1.2: CLAP Flowchart

A_v = vapor flow area

ρ_v = vapor density

w = vapor mass flowrate

$\dot{m}_v$ = the rate of vapor generation per unit length of channel

$\frac{\partial p}{\partial z}$ = channel axial pressure gradient

F_v = cladding/vapor interfacial friction force per unit volume of vapor.

The friction force term is evaluated from the following equation:

$$F_v = \frac{f_{sf} M w |w|}{2\rho_v A_v^2 D_v} \tag{13.2-3}$$

where

f_{sf} = single-phase friction factor

M = flooding multiplier

D_v = hydraulic diameter for the vapor.

The hydraulic diameter is estimated by

$$D_v = D_h \sqrt{\alpha} \tag{13.2-4}$$

where

D_h = hydraulic diameter for bare fuel or fuel pin (depending upon the zone)

α = vapor fraction based on area available for molten steel plus vapor.

This equation assumes that the hydraulic diameter D_v varies as the square root of the area $_v$ (which would be exact for a circular vapor flow passage) and hence scales as the square root of vapor fraction (proportional to the flow area).

The smooth wall (single-phase) friction factor is given by

$$f_{sf} = \text{AFRV} \left(\text{Re}\right)_v^{\text{BFRV}} \tag{13.2-5}$$

$$\left(\text{Re}\right)_v = w \frac{D_v}{A_v} \mu_v \tag{13.2-6}$$

where

AFRV, BFRV = input constants

μ_v = vapor viscosity.

For consistency, the inputed constants AFRV and BFRV used in CLAP are identical to those used in the sodium voiding model.

The flooding multiplier is flow-regime dependent. The transition from smooth wall friction $(M = 1)$ to a chaotic two-phase flow structure ($M = M_{2\Phi}$, where $M_{2\Phi}$ is defined later) occurs nominally when the vapor velocity exceeds the flooding velocity. The current version of CLAP uses the correlation of Fauske, et al. [13-4]:

$$v_{\text{flood}} = \left[\frac{32}{3} \frac{\rho_c g \Delta r_c}{\rho_v f_{sf} M_{2\Phi}} \right]^{1/2} \tag{13.2-7}$$

where

Δr_c = one-half the molten cladding thickness

ρ_c = molten cladding density.

The CLAP model has a hysteresis effect in the treatment of the flooding multiplier whereby flooding and the inverse, deflooding, occur at slightly different velocities as follows:

$$\text{set} \quad M^n = M_{2\Phi}, \quad \text{if} \quad M^{n-1} = 1 \quad \text{and} \quad \left|\frac{w}{A_v}\rho_v\right| > 1.1 v_{\text{flood}} \tag{13.2-8}$$

$$\text{set} \quad M^n = 1, \quad \text{if} \quad M^{n-1} = M_{2\Phi} \quad \text{and} \quad \left|\frac{w}{A_v}\rho_v\right| < 0.9 v_{\text{flood}} \tag{13.2-9}$$

where n is the current time-step number. If the conditions in either Eq. (13.2-8) or Eq. (13.2-9) are not met, then the previous value of M is retained.

The two-phase friction multiplier is given by

$$M_{2\Phi} = I \cdot [1 + \varepsilon (1 - \alpha)] \quad \text{for } \alpha > \alpha_{\text{crit}} \tag{13.2-10}$$

$$M_{2\Phi} = I \cdot [1 + \varepsilon (1 - \alpha_{\text{crit}})] \quad \text{for } \alpha \leq \alpha_{\text{crit}} \tag{13.2-11}$$

where ε and α_{crit} are input and I is an incoherence multiplier. For values of $\varepsilon = 75$ and $\alpha_{\text{crit}} = 0$, Eq. (13.2-10) is equivalent to the Wallis correlation [13-5].

Radial incoherency in cladding melting, which occurs during the early portion of cladding motion, results in a lower bundle pressure drop and lower cladding velocity than predicted by one-dimensional models that ignore this effect [13-6]. The incoherence multiplier allows for this incoherency effect by temporarily reducing the two-phase friction multiplier during early cladding motion in the following manner:

$$I = [\text{fps}(\text{fps})_0]^x \quad \text{for fps} < (\text{fps})_0 \tag{13.2-12}$$

$$I = 1 \quad \text{for fps} \geq (\text{fps})_0 \tag{13.2-13}$$

where fps are full-power seconds since cladding first moved in a channel and $(\text{fps})_0$ and x are input constants.

The code sets $M_{2\phi}$ equal to unity, equivalent to a smooth wall friction factor, if the value of $M_{2\phi}$ computed by Eq. (13.2-10) is less than unity. Consequently, the effect of the multiplier "I" is to delay flooding by $(\text{fps})_0$ full-power seconds. For cases where the vapor velocity is below that necessary for flooding, neither $(\text{fps})_0$ nor ε will have any effect on the cladding motion. Values for $(\text{fps})_0$ and x of 0.3 and 3 are provisionally recommended; better values will be determined later by calibration of the CLAP model using experimental data.

13.2.2 Moving Cladding Basic Equations

The mass, momentum, and energy conservation equations for moving (molten) cladding are

$$\frac{\partial}{\partial t}(\rho_c A_c) + \frac{\partial}{\partial z}(\rho_c A_c v_c) = \dot{m}_c \tag{13.2-14}$$

$$\frac{\partial}{\partial t}(\rho_c A_c c_c) + \frac{\partial}{\partial z}\left(\rho_c A_c v_c^2\right) + A_c \frac{\partial p}{\partial z} + A_c F_p - A_v F_v + A_c \rho_c g = \begin{cases} \dot{m}_c v_c & \text{if freezing} \quad \text{(13.2-13a)} \\ 0, & \text{if melting} \quad \text{(13.2-13b)} \end{cases} \tag{13.2-15}$$

$$\frac{\partial}{\partial t}(\rho_c A_c e_c) + \frac{\partial}{\partial z}(\rho_c A_c v_c e_c) = \phi_c P_r + \dot{m}_c e_c \tag{13.2-16}$$

A_c = moving cladding cross-sectional area

v_c = moving cladding velocity

$\dot{m}_c$ = mass rate of melting of substrate per unit length of channel (or rate of freezing if negative)

F_p = pin friction-force per unit volume of molten cladding

e_c = moving cladding internal energy

ϕ_c = flux of sensible heat from the solid/liquid interface into the molten cladding

P_r = perimeter of solid/liquid interface.

The pin or bare-fuel friction force is evaluated by the following:

$$F_p = \frac{c_f \rho_c v_c |v_c|}{2D_c} \tag{13.2-17}$$

where

c_f = pin friction factor

D_c = molten cladding hydraulic diameter.

The CLAZAS [13-3] formulation for the friction factor has been incorporated into CLAP:

$$c_f = \frac{64}{\mathrm{Re}} \quad \text{for Re} < (\mathrm{Re})_{\text{break}} \tag{13.2-18}$$

$$c_f = b_f \quad \text{for Re} \geq (\mathrm{Re})_{\text{break}} \tag{13.2-19}$$

where Re is the molten-cladding Reynolds number defined by:

$$\mathrm{Re} = \frac{D_c \rho_c |v_c|}{\mu_c} \tag{13.2-20}$$

$(\mathrm{Re})_{\text{break}}$ is the critical Reynolds number for the transition from laminar to turbulent friction and b_f is the turbulent Moody friction factor. The CLAZAS derived values for $(\mathrm{Re})_{\text{break}}$ and b_f of 2.1×10^3 and 2.0×10^{-2} are currently recommended as input values.

The cladding viscosity depends upon the cladding temperature T_c and melt-fraction f as follows

$$\mu_c = \mu_m \exp\left(\frac{a}{T_c} - \frac{a}{T_m}\right) \quad \text{for } T_c > T_m \tag{13.2-21}$$

$$\mu_c = (\mu_t - \mu_m)(1-f)^q + \mu_m \quad \text{for } T_c = T_m \tag{13.2-22}$$

$$\mu_c = \mu_s \quad \text{for } T_c < T_m \tag{13.2-23}$$

where T_m is the cladding melting temperature, μ_s, μ_t, μ_m are the user-supplied viscosities for solid cladding, at the solidus temperature, and cladding at the liquidus temperature, and a and q are input constants. Values for μ_m and a of 6.42×10^{-3} Pa-s and 5492 K are recommended [13-7]. The input parameters μ_t, μ_s, and q are not properties per se, but are constants in a phenomenological model and hence need to be determined by calibration with cladding motion test data. In general, the values selected for μ_t and μ_s should be sufficiently large to effectively arrest cladding motion for solid cladding. Provisional values for both μ_t and μ_s of 10^4 Pa-s (10^5 poise), from the CLAZAS sample problem [13-3], and for q of 0.5 are recommended.

The momentum and energy Eq. (13.2-15) and Eq. (13.2-16) are converted to non-conservative form using Eq. (13.2-14), giving the following:

$$\rho_c \left[\frac{\partial v_c}{\partial t} + v_c \frac{\partial v_c}{\partial t}\right] + \frac{\partial p}{\partial z} + F_p - \frac{A_v}{A_c} F_v + g\rho_c = \begin{cases} 0 & \text{if freezing} \quad \text{(13.2-19a)} \\ -m_c \dfrac{\dot{v}_c}{A_c} & \text{if melting} \quad \text{(13.2-19b)} \end{cases} \tag{13.2-24}$$

$$\frac{\partial e_c}{\partial t} + v_c \frac{\partial e_c}{\partial z} = \frac{\phi_c P_r}{A_c \rho_c} \tag{13.2-25}$$

13.2.3 Refrozen Cladding Basic Equations

The conservation equations for the refrozen steel layer are

$$\frac{\partial}{\partial \mathrm{t}}\left(\rho_{\mathrm{s}} A_{\mathrm{s}}\right) = -\dot{m}_{\mathrm{c}} \tag{13.2-26}$$

$$\frac{\partial}{\partial \mathrm{t}}\left(\rho_{\mathrm{s}} A_{\mathrm{s}} e_{\mathrm{s}}\right) = \phi_{\mathrm{r}} P_{\mathrm{e}} - \phi P_{\mathrm{r}} - \dot{m}_{\mathrm{e}} e_{\mathrm{s}} \tag{13.2-27}$$

where

ρ_{s} = refrozen steel density

A_{s} = refrozen steel cross-sectional area

e_{s} = refrozen steel internal energy

ϕ_{r} = heat flux at the interface between intact cladding and refrozen cladding

P_{e} = outer perimeter of intact cladding

ϕ = flux of sensible heat from the refrozen cladding to the melt (solid/liquid) interface.

Recall that the quantity $\dot{m}_{\mathrm{c}}$ was the rate of melting (negative for freezing) and hence appears as a sink (negative sign) in the refrozen cladding Eq. (13.2-26) and Eq. (13.2-27).

By manipulating the preceding equations, we obtain the nonconservative form of the energy equation:

$$\frac{\partial \mathrm{e_s}}{\partial \mathrm{t}} = \frac{\left(\phi_{\mathrm{r}} P_{\mathrm{e}} - \phi P_{\mathrm{r}}\right)}{\rho_{\mathrm{s}} A_{\mathrm{s}}} \tag{13.2-28}$$

Note that the sensible heat fluxes on either side of the melt layer are not the same (see Figure 13.2.1). The difference ϕ_{hf} is the heat flux for fusion, defined by

$$\phi_{\mathrm{hf}} = \phi - \phi_{\mathrm{c}} \tag{13.2-29}$$

The rate of melting (or freezing, if negative) is directly related to the fusion heat flux:

$$\dot{m}_{\mathrm{c}} = P_{\mathrm{r}} \frac{\phi_{hf}}{\lambda} \tag{13.2-30}$$

where λ is an effective heat of fusion of the cladding defined as

$$\lambda = e_{\mathrm{c}} - e_{\mathrm{s}} \tag{13.2-31}$$

13.2.4 Heat-transfer Relationships

13.2.4.1 Moving Cladding on Bare Fuel

The configuration in the molten zone is shown in Figure 13.2.1. There is no solid cladding in this region and, therefore, the heat flux for melting, ϕ_{hf}, is zero. The sensible heat flux is given by

$$\phi_{\mathrm{c}} = h\left(T_{\mathrm{f}} - T_{\mathrm{c}}\right) \tag{13.2-32}$$

where

T_{f} = fuel pellet surface temperature

h = coefficient of heat transfer from the fuel to molten cladding.

The heat-transfer coefficient for the moving cladding is defined from a correlation for liquid-metal heat transport as

$$\frac{hD_c}{k_c} = C_1 \left(D_c \rho_c c_{pc} \frac{|v_c|}{k_c} \right)^{C_2} + C_3 \tag{13.2-33}$$

where

k_c = molten cladding thermal conductivity

c_{pc} = molten cladding specific heat capacity

C_1, C_2, C_3 = input constants.

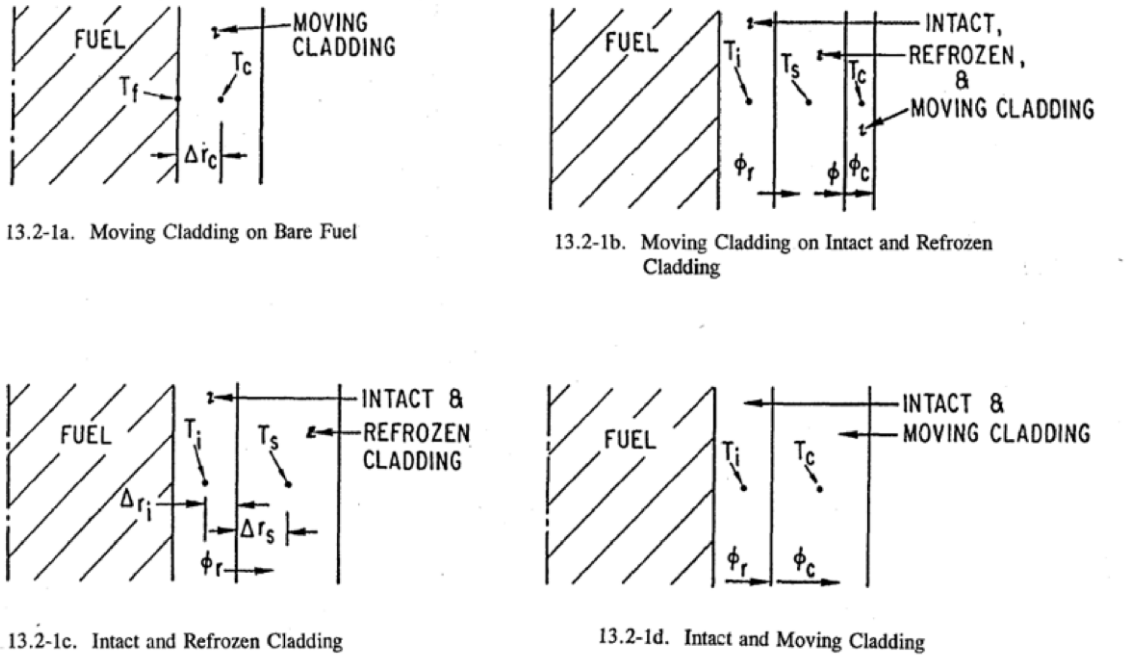


Figure 13.2.1: Schematic Showing the Different Heat-Transfer Configuration in CLAP

The same correlation and constants are used for computing heat-transfer coefficients for single-phase sodium flow in the channel, (Eq. (3.3-9) of Chapter 3). The correlation of Maresca and Dwyer [13-8] for in-line flow in rod bundles is recommended, for which the constants are:

$$C_1 = 0.0155 \left(\overline{\psi} \right)^{0.86} \tag{13.2-34}$$

$$C_1 = 0.86 \tag{13.2-35}$$

$$C_3 = 6.66 + 3.126 \left(\frac{P}{D} \right) + 1.184 \left(\frac{P}{D} \right)^2 \tag{13.2-36}$$

where $\overline{\psi}$ = mean ratio of the thermal and momentum diffusivities, and $\frac{P}{D}$ = rod pitch-to-diameter ratio. Since the code does not allow for flow-rate-dependent values for C_1, it is suggested that $\overline{\psi}$ be set equal to unity in Eq. (13.2-34) with the understanding that the resulting heat-transfer coefficients will be slightly overestimated.

13.2.4.2 Moving Cladding on Intact and Refrozen Cladding

Various configurations can exist in the region outside the molten zone, depending upon whether or not moving cladding and/or refrozen cladding are present in addition to the original intact solid cladding. In this subsection, we consider the case where all three components are present (see Figure 13.2.1). The various heat fluxes shown in the figure are given by

$$\phi_{\mathrm{r}} = k_{\mathrm{c}} \frac{(T_{\mathrm{i}} - T_{\mathrm{s}})}{(\Delta r_{\mathrm{i}} + \Delta r_{\mathrm{s}})} \tag{13.2-37}$$

$$\phi_{\mathrm{r}} = k_{\mathrm{c}} \frac{(T_{\mathrm{s}} - T_{\mathrm{m}})}{\Delta r_{\mathrm{s}}} \tag{13.2-38}$$

$$\phi_{\mathrm{c}} = h\,(T_{\mathrm{m}} - T_{\mathrm{c}}) \tag{13.2-39}$$

where

T_{i} = temperature of the intact solid cladding

T_{s} = temperature of the refrozen cladding

T_{m} = cladding melting temperature

Δr_{i} = half-thickness of the intact solid cladding

Δr_{s} = half-thickness of the refrozen cladding.

The film coefficient formula is identical to Eq. (13.2-33) and the fusion heat flux ϕ_{hf} is given by Eq. (13.2-29).

13.2.4.3 Intact and Refrozen Cladding

In the absence of moving cladding (see Figure 13.2.1), we set $\phi = 0$ and $\phi_{\mathrm{hf}} = 0$. The heat transfer between the initial and refrozen cladding is computed from

$$\phi_{\mathrm{r}} = k_{\mathrm{c}} \frac{(T_{\mathrm{i}} - T_{\mathrm{s}})}{(\Delta r_{\mathrm{i}} + \Delta r_{\mathrm{s}})} \tag{13.2-40}$$

which is identical to Eq. (13.2-37).

13.2.4.4 Intact and Moving Cladding

The moving cladding may be sufficiently hot to melt through the layer of refrozen cladding; this results in the configuration (see Figure 13.2.1) in which just the intact cladding and moving cladding are present. In the numerical model this situation also arises when moving cladding first contacts the intact cladding in an axial segment, in which case refrozen cladding is formed and will be present in the next time step.

The heat fluxes ϕ_1 and ϕ_2 and trial heat flux are calculated as follows

$$\phi_1 = k_{\mathrm{c}}\,(T_{\mathrm{i}} - T_{\mathrm{m}})\,\Delta r_{\mathrm{i}} \tag{13.2-41}$$

$$\phi_2 = h\,(T_{\mathrm{m}} - T_{\mathrm{c}}) \tag{13.2-42}$$

$$\phi_{\mathrm{trial}} = \phi_1 - \phi_2 \tag{13.2-43}$$

Recalling that ϕ_{r} is the outer surface heat flux for the initial cladding, ϕ_{c} the sensible heat flux for moving cladding, and ϕ_{hf} the latent heat flux associated with refrozen cladding, we have

$$\phi_{\mathrm{hf}} = 0, \phi_r = \phi_2, \text{ and } \phi_{\mathrm{c}} = \phi_2 \quad \text{for } \phi_{\mathrm{trial}} \geq 0 \tag{13.2-44}$$

$$\phi_{\mathrm{hf}} = \phi_{\mathrm{trial}}, \phi_{\mathrm{r}} = \phi_1, \text{ and } \phi_{\mathrm{c}} = \phi_2 \quad \text{for } \phi_{\mathrm{trial}} < 0 \tag{13.2-45}$$

13.2.5 Cladding Physical Properties Relationships

The density of the solid and liquid cladding are assumed to be linear functions of temperature:

$$\rho_{\mathrm{s}} = \rho_{\mathrm{s}}^{0}\left[1 - 3\beta\left(T - T_{\mathrm{ref}}\right)\right] \tag{13.2-46}$$

$$\rho_{\mathrm{c}} = \rho_{\mathrm{c}}^{0}\left[1 - C\left(T - T_{\mathrm{m}}\right)\right] \tag{13.2-47}$$

where

ρ_{s}^{0} = density of solid cladding at reference temperature T_{ref}

ρ_{c}^{0} = density of cladding at the liquidus temperature

β = linear coefficient of thermal expansion for solid

C = volumetric coefficient of thermal expansion for liquid.

For partly-molten cladding, the density is evaluated using

$$\rho = \left[\frac{(1-f)}{\rho_{\mathrm{s}}}\left(T_{\mathrm{m}}\right) + \frac{f}{\rho_{\mathrm{c}}}\right]^{-1} \tag{13.2-48}$$

where f is the melt fraction as defined by Eq. (13.2-51).

The internal energies of the solid and molten cladding are also assumed to be linear functions of temperature

$$e_{\mathrm{s}} = c_{\mathrm{ps}}T \tag{13.2-49}$$

$$e_{\mathrm{c}} = e_{\mathrm{c}}^{o} + c_{\mathrm{pc}}T \tag{13.2-50}$$

where $e_{\mathrm{c}}^{o} = \lambda^{o} + \left(C_{\mathrm{ps}} - C_{\mathrm{pc}}\right)T_{\mathrm{m}}$, and math:*lambda^{o}* = heat of fusion. For mixtures, the mass melt fraction is given by

$$f = \frac{\left(e - c_{\mathrm{ps}}T_{\mathrm{m}}\right)}{\lambda^{o}} \tag{13.2-51}$$

13.3 Method of Solution

13.3.1 CLAP Initialization and Cladding Meltthrough

The temperature and melt fraction of initial cladding are computed after the onset of boiling by the SAS fuel-pin model located in the subroutine TSHTRV and the calculation continues after CLAP initiation. This subroutine also contains the logical statements for CLAP initiation and cladding melt-through.

The first cladding to move is assumed to be the first axial segment that is completely molten; CLAP is initiated with this first melt-through. Any other axial segments join the molten cladding region if any one of its three radial segment nodes is molten; i.e., is above the liquidus temperature.

When entering CLAP (which uses coolant time steps) from a heat-transfer time step, a check is made to see whether any previously solid cladding nodes are now to be treated as molten cladding. If so, the total mass and energy of the initial, refrozen, and moving cladding are evaluated and these become the mass and energy of the moving cladding.

13.3.2 Solution of the Cladding Energy Equations

13.3.2.1 Interface with the Fuel-pin Model

There are slight differences in the thermodynamic representations of molten steel in the fuel-pin and CLAP models. In particular, the fuel-pin model in subroutine TSHTRV has a melting band (discrete solidus and liquidus temperatures) and a temperature-dependent specific heat, whereas CLAP has a single melting temperature and constant specific heats. If cladding temperatures from the fuel-pin model were used directly in CLAP calculations (for example, in adding segments to the molten region), then, possibly, energy would not be conserved due to differences in the thermodynamic models. To insure energy conservation, CLAP instead uses nodal energies rather than temperatures. A mean energy is computed from the SAS nodal energies:

$$\overline{e}_{\mathrm{j}} = \frac{1}{4} e_{1,\mathrm{j}} + \frac{1}{2} e_{2,\mathrm{j}} + \frac{1}{4} e_{3,\mathrm{j}} \tag{13.3-1}$$

where the $e_{\mathrm{i,j}}$'s are the energies of the three radial nodes in axial segment j. CLAP then evaluates the intact-cladding temperature T_{i} and melt fraction using $\overline{e}$ and Eq. (13.2-49) to Eq. (13.2-51).

The fuel-pin model computes fuel temperatures in the molten cladding zone and both fuel and intact cladding temperatures for the remainder of the fueled section. For the molten zone, the CLAP routine integrates the fuel surface heat loss for each segment for a heat-transfer time interval:

$$Q_{\mathrm{j}} = P_{\mathrm{r}} \Delta z_{\mathrm{j}} \int_{\mathrm{t}^*}^{t^*+\Delta t^*} \phi_{\mathrm{c,j}} \mathrm{dt} \tag{13.3-2}$$

where

$\Delta z_{\mathrm{j}} = z_{\mathrm{j+1}} - z_{\mathrm{j}}$

t^* is time at start of current heat-transfer time step

Δt^* is current heat-transfer time interval

ϕ_{c} is the heat flux given by Eq. (13.2-32)

z_{j} is the elevation at the bottom of the "j"th segment.

This heat loss is utilized at the outer fuel boundary condition by the fuel-pin model. In a similar fashion, the heat loss from the outer surface of an intact cladding segment is computed by

$$Q_{\mathrm{j}} = P_{\mathrm{e}} \Delta z_{\mathrm{j}} \int_{\mathrm{t}^*}^{t^*+\Delta t^*} \phi_{\mathrm{r,j}} \mathrm{dt} \tag{13.3-3}$$

where ϕ_{r} is the heat flux from Eq. (13.2-37) or Eq. (13.2-44), depending upon the configuration. This heat loss becomes part of the outer cladding boundary condition in the fuel-pin model, being added to the coolant heat loss, if any.

13.3.2.2 Intact and Refrozen Cladding

For this configuration, we set φ = 0 (adiabatic outer boundary) in the refrozen cladding energy Eq. (13.2-28). The energy equation is then combined with Eq. (13.2-37) and Eq. (13.2-49) to give

$$\frac{\mathrm{dT_{s,j}}}{\mathrm{dt}} = \xi_1 \left(T_{\mathrm{i}} - T_{\mathrm{s}} \right) \Big|_{\mathrm{j}} \tag{13.3-4}$$

where

$$\xi_{1,\mathrm{j}} = \frac{P_{\mathrm{e}} k_{\mathrm{c}}}{\left[A_{\mathrm{s}} \rho_{\mathrm{s}} c_{\mathrm{ps}} \left(\Delta r_{\mathrm{s}} + \Delta r_{\mathrm{i}} \right) \right]} \Bigg|_{\mathrm{j}}$$

Since Eq. (13.2-28) has been reduced from a general field equation to a nodal equation, it becomes an ordinary differential equation (in time) in the process. The subscript j, of course, denotes the axial segment number for which the parameter is evaluated.

Eq. (13.3-4) is converted to an implicit finite difference equation by substituting for the left side, the following:

$$\frac{\mathrm{dT_{s,j}}}{\mathrm{dt}} \approx \frac{T^n_{\mathrm{s,j}}}{d} \tag{13.3-5}$$

and by evaluating $T_{\mathrm{s,j}}$ on the right side at the advanced time

$$T^{n+1}_{\mathrm{s,j}} = T^n_{\mathrm{s,j}} + \Delta T^n_{\mathrm{s,j}} \tag{13.3-6}$$

giving (after some rearrangement):

$$T^n_{\mathrm{s,j}} = \left.\frac{\xi_1 \left(T_\mathrm{i} - T_\mathrm{s}\right) t}{1 + \xi_1 t}\right|^n_{\mathrm{j}} \tag{13.3-7}$$

The implicit form is used rather than an explicit solution because it is stable for large ξ_1 (or small A_s) and a reasonably sized Δt.

The heat flux ϕ_r, required for the fuel-pin boundary condition Eq. (13.3-3) is evaluated from the energy equation:

$$\phi^n_{\mathrm{r,j}} = \left.\frac{A_\mathrm{s}\rho_\mathrm{s}C_{\mathrm{ps}}\Delta T_\mathrm{s}}{P_\mathrm{e}\Delta t}\right|^n_{\mathrm{j}} \tag{13.3-8}$$

13.3.2.3 Evaluation of Convective Term in the Moving Cladding Energy Equation

The CLAP model uses a modified donor cell technique developed by Bohl [13-2] for the evaluation of the convective term

$$C_\mathrm{v} = A_\mathrm{c}\rho_\mathrm{c}v_\mathrm{c}\frac{\partial \mathrm{e_c}}{\partial \mathrm{z}} \tag{13.3-9}$$

in Eq. (13.2-25). This scheme uses a segment boundary mass flow w^* which is estimated by

$$w^*_\mathrm{j} = \frac{1}{2}\left[A_{\mathrm{c,j-1}}\rho_{\mathrm{c,j-1}}v_{\mathrm{c,j-1}} + A_{\mathrm{c,j}}\rho_{\mathrm{c,j}}v_{\mathrm{c,j}}\right] \tag{13.3-10}$$

The flow direction is determined by a mean flow w_m:

$$w_{\mathrm{m,j}} = \frac{1}{2}\left(\mathrm{w}^*_\mathrm{j} + w^*_{\mathrm{j+1}}\right) \tag{13.3-11}$$

The recipe for the convective term for a unblocked segment is as follows

$$C_{\mathrm{v,j}} = w^*_\mathrm{j}\frac{e_{\mathrm{c,j}} - e_{\mathrm{c,j-1}}}{z_{\mathrm{m,j}} - z_{\mathrm{m,j-1}}} \quad \text{for } w_{\mathrm{m,j}} > 0, w^*_\mathrm{j} > 0 \tag{13.3-12}$$

$$C_{\mathrm{v,j}} = w^*_{\mathrm{j+1}}\frac{e_{\mathrm{c,j+1}} - e_{\mathrm{c,j}}}{z_{\mathrm{m,j+1}} - z_{\mathrm{m,j}}} \quad \text{for } w_{\mathrm{m,j}} < 0, w^*_{\mathrm{j+1}} < 0 \tag{13.3-13}$$

$$C_{\mathrm{v,j}} = 0, \quad \text{otherwise} \tag{13.3-14}$$

The parameter $z_{\mathrm{m,j}}$ is the nodal elevation, which is related to the segment interface elevations z_j by

$$z_{\mathrm{m,j}} = \frac{1}{2}\left(z_\mathrm{j} + z_{\mathrm{j+1}}\right) \tag{13.3-15}$$

If the "j"th segment is nearly blocked, then the energy convection computed by Eq. (13.3-12) may be too large in magnitude. In this case, we define a segment flow w as

$$w_{\mathrm{j}} = A_{\mathrm{c,j}} \rho_{\mathrm{c,j}} v_{\mathrm{c,j}} \tag{13.3-16}$$

and compute the convective terms by (blocked segment only)

$$C_{\mathrm{v,j}} = w_{\mathrm{j}} \frac{e_{\mathrm{c,j}} - e_{\mathrm{c,j-1}}}{z_{\mathrm{m,j}} - z_{\mathrm{m,j-1}}} \quad \text{for } w_{\mathrm{m,j}} > 0, w_{\mathrm{j}} > 0 \tag{13.3-17}$$

$$C_{\mathrm{v,j}} = w_{\mathrm{j}} \frac{e_{\mathrm{c,j+1}} - e_{\mathrm{c,j}}}{z_{\mathrm{m,j+1}} - z_{\mathrm{m,j}}} \quad \text{for } w_{\mathrm{mj}} < 0, w_{\mathrm{j}} > 0 \tag{13.3-18}$$

$$C_{\mathrm{v,j}} = 0, \quad \text{otherwise} \tag{13.3-19}$$

The scheme also suppresses convection (sets $C_{\mathrm{v}} = 0$) if the adjoining donor cell (either $j-1$ or $j+1$ depending on the sign of w_{m}) is blocked.

13.3.2.4 Moving Cladding on Bare Fuel

The outermost radial fuel node in the fuel-pin model is duplicated in CLAP for axial segments within the molten zone. The outer fuel temperature in CLAP, T_{f}, is reset to the value computed by the pin model at the start of each heat-transfer time step. However, between heat transfer time steps, the fuel temperature is computed in CLAP; this is done primarily to achieve numerical stability, but it may also improve the accuracy of results. The configuration of the SAS outer fuel node and moving cladding is shown in Figure 13.3.1. The transient heat-balance equation for this outer node is

$$\pi \left(r_{\mathrm{NR}}^2 - r_{\mathrm{NT}}^2\right) \rho_{\mathrm{f}} c_{\mathrm{pf}} \frac{\partial \mathrm{T_f}}{\partial \mathrm{t}} + 2\pi_{\mathrm{NT}} k_{\mathrm{f}} \left.\frac{\partial \mathrm{T}}{\partial \mathrm{r}}\right|_{\mathrm{r_{NT}}} - \pi \left(r_{\mathrm{NR}}^2 - r_{\mathrm{NT}}^2\right) Q_{\mathrm{NT}} = -P_{\mathrm{r}} h \left(T_{\mathrm{f}} - T_{\mathrm{c}}\right) \tag{13.3-20}$$

where reference to axial core segment j is tacitly implied.

It is assumed that the heat flow from the interior fuel nodes and the heat generated in the outer fuel node (given by the second and third terms in Eq. (13.3-20)) are relatively constant over one heat-transfer time step. We can therefore approximate these terms by evaluating them at the beginning of a heat-transfer time step. These terms are represented symbolically by the constant C_{f}, defined by

$$C_{\mathrm{f}} = 2\pi r_{\mathrm{NT}} k_{\mathrm{f}} \left.\frac{\partial \mathrm{T}}{\partial \mathrm{r}}\right|_{\mathrm{r_{NT}}} - \pi \left(r_{\mathrm{NR}}^2 - r_{\mathrm{NT}}^2\right) Q_{\mathrm{NT}}, \quad \text{at } t = t^* \tag{13.3-21}$$

where t^* is the beginning of the heat-transfer time step.

Because of the high heat-transfer coefficient to the molten cladding Eq. (13.2-33), the fuel-surface temperature and molten cladding temperature are closely coupled. For that reason, the term representing the instantaneous heat loss to the molten steel (on the right side of Eq. (13.3-20)) is retained as is, rather than incorporating it into the constant C_{f}. Combining Eq. (13.3-20) and Eq. (13.3-21) and rearranging, we have

$$\frac{\mathrm{dT_{f,j}}}{\mathrm{dt}} + \gamma_{\mathrm{f}} = \left. -\xi_{\mathrm{f}} \left(T_{\mathrm{f}} - T_{\mathrm{c}}\right) \right|_{\mathrm{j}} \tag{13.3-22}$$

where

$$\gamma_{\mathrm{f,j}} = \left. \frac{C_{\mathrm{f}}}{\left[\pi \left(r_{\mathrm{NR}}^2 - r_{\mathrm{NT}}^2\right)\right]} \rho_{\mathrm{f}} c_{\mathrm{pf}} \right|_{\mathrm{j}} \tag{13.3-23}$$

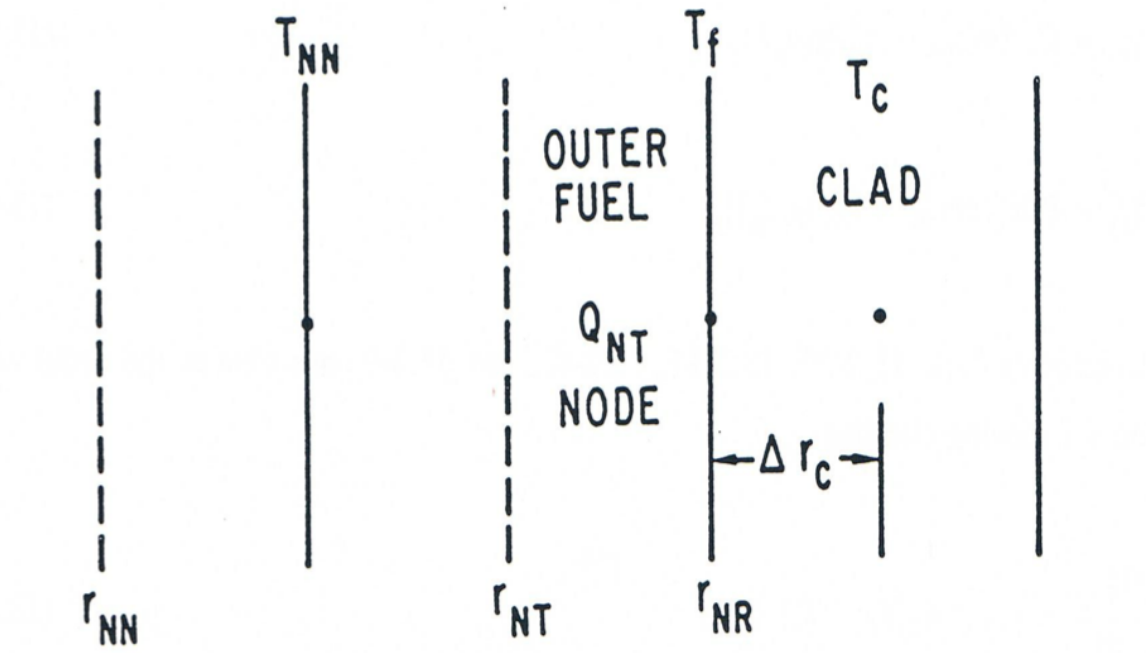


Figure 13.3.1: Schematic Showing the Interface of CLAP with the SAS Fuel Model

$$\xi_{\mathrm{f,j}} = \left. \frac{P_{\mathrm{r}} h}{\left[\pi \left(r_{\mathrm{NR}}^2 - r_{\mathrm{NT}}^2\right) \rho_{\mathrm{f}} c_{\mathrm{pf}}\right]} \right|_{\mathrm{j}} \tag{13.3-24}$$

Combining Eq. (13.2-25), Eq. (13.2-32), Eq. (13.2-50), and Eq. (13.3-9), one obtains the nodal energy equation for moving cladding:

$$\left. \frac{\mathrm{dT_{c,j}}}{\mathrm{dt}} + \gamma_{\mathrm{c,j}} = \xi_2 \left(T_{\mathrm{f}} - T_{\mathrm{c}}\right) \right|_{\mathrm{j}} \tag{13.3-25}$$

where

$$\gamma_{\mathrm{c,j}} = \left. \frac{C_{\mathrm{v}}}{\left(A_{\mathrm{c}} \rho_{\mathrm{c}} c_{\mathrm{pc}}\right)} \right|_{\mathrm{j}} \tag{13.3-26}$$

$$\xi_{2,\mathrm{j}} = \left. \frac{P_{\mathrm{r}} h}{\left(A_{\mathrm{c}} \rho_{\mathrm{c}} c_{\mathrm{pc}}\right)} \right|_{\mathrm{j}} \tag{13.3-27}$$

Eq. (13.3-22) and Eq. (13.3-25) are converted to time-implicit difference equations by approximating the time derivatives according to

$$\frac{\mathrm{dT_{f,j}}}{\mathrm{dt}} \simeq \frac{\Delta T_{\mathrm{f,j}}^n}{\Delta t^n} \tag{13.3-28}$$

$$\frac{\mathrm{dT_{c,j}}}{\mathrm{dt}} \simeq \frac{\Delta T_{\mathrm{c,j}}^n}{\Delta t^n} \tag{13.3-29}$$

and evaluating the temperatures on the right-hand side at the advanced time t^{n+1}. After substituting

$$T_{\mathrm{c,j}}^{n+1} = T_{\mathrm{f,j}}^n + \Delta T_{\mathrm{f,j}}^n \tag{13.3-30}$$

$$T_{\mathrm{c,j}}^{n+1} = T_{\mathrm{c,j}}^n + \Delta T_{\mathrm{c,j}}^n \,, \tag{13.3-31}$$

the two difference equations are solved simultaneously for ΔT_{f} and ΔT_{c} and second-order terms in Δt are discarded, with the following results

$$\Delta T_{\mathrm{c,j}}^n = \left. -\frac{\left[\xi_2 \left(T_{\mathrm{f}} - T_{\mathrm{c}}\right) - \gamma_{\mathrm{c}}\right] \Delta t}{1 + \left(\xi_2 + \xi_{\mathrm{f}}\right) \Delta t} \right|_{\mathrm{j}}^n \tag{13.3-32}$$

$$\Delta T_{\mathrm{f,j}}^n = \left. -\frac{\left[\xi_{\mathrm{f}} \left(T_{\mathrm{f}} - T_{\mathrm{c}}\right) - \gamma_{\mathrm{f}}\right] \Delta t}{1 + \left(\xi_2 + \xi_{\mathrm{f}}\right) \Delta t} \right|_{\mathrm{j}}^n \tag{13.3-33}$$

The heat flux ϕ_{c}, which is needed to evaluate the integral in Eq. (13.3-2), is computed using a finite difference form of the moving clad energy Eq. (13.2-25):

$$\phi_{\mathrm{c}} = \frac{C_{\mathrm{v}}}{P_{\mathrm{r}}} + \frac{A_{\mathrm{c}} \rho_{\mathrm{c}} c_{\mathrm{pc}} \Delta T_{\mathrm{c}}}{P_{\mathrm{r}} \Delta t} \tag{13.3-34}$$

13.3.2.5 Moving Cladding on Intact and Refrozen Cladding

The nodal energy equation for the moving cladding for this configuration is obtained by combining Eq. (13.2-25), Eq. (13.2-39), Eq. (13.2-50), and Eq. (13.3-9), giving

$$\left. \frac{\mathrm{dT_{c,j}}}{\mathrm{dt}} + \gamma_{\mathrm{c}} = \xi_2 \left(T_{\mathrm{m}} - T_{\mathrm{c}}\right) \right|_{\mathrm{j}} \tag{13.3-35}$$

where γ_c and ξ_2 are defined in Eq. (13.3-26) and Eq. (13.3-27).

Solving this equation implicitly, i.e., replacing T_c by the updated value, approximating the time derivative by Eq. (13.3-29), and solving for ΔT_c, we get

$$\Delta T_{c,j}^n = -\frac{\left[\xi_2 \left(T_m - T_c\right) - \gamma_c\right] \Delta t}{1 + \xi_2 \Delta t}\Bigg|_j^n \tag{13.3-36}$$

The equation for the refrozen cladding temperature is found by substituting Eq. (13.2-37), Eq. (13.2-38), and Eq. (13.2-49) into the energy Eq. (13.2-28), resulting in

$$\Delta T_{c,j}^n = -\frac{\left[\xi_2 \left(T_m - T_c\right) - \gamma_c\right] \Delta t}{1 + \xi_2 \Delta t}\Bigg|_j^n \tag{13.3-37}$$

where

$$\xi_3 = \frac{P_r k_c}{A_s \rho_s c_{ps} \Delta r_s} \tag{13.3-38}$$

Converting Eq. (13.3-37) to a finite-difference, implicit equation and solving for ΔT_s, one obtains

$$\Delta T_{s,j}^n = -\frac{\left[\xi_1 \left(T_i - T_s\right) - \xi_3 \left(T_s - T_m\right)\right] \Delta t}{1 + \left(\xi_1 + \xi_3\right) \Delta t}\Bigg|_j^n \tag{13.3-39}$$

The heat fluxes ϕ_r and ϕ_{hf} are required for the fuel-pin boundary conditions Eq. (13.3-3) and to evaluate the rate of melting Eq. (13.2-30). These are evaluated using discretized versions of the energy Eq. (13.2-25) and Eq. (13.2-28) after computing ϕ_r from Eq. (13.2-37). Then ϕ_{hf} is computed using Eq. (13.2-29).

Several constraints apply to ϕ_{hf}. For positive ϕ_{hf} (melting), more cladding cannot melt than actually exists in a frozen state. This restriction can be expressed as

$$\phi_{hf} \leq \frac{A_s \rho_s \lambda_c}{\Delta t P_r} \tag{13.3-40}$$

In case ϕ_{hf} is adjusted to satisfy Eq. (13.3-40), new values of ϕ, ϕ_c and ϕ_r are computed from discretized versions of energy Eq. (13.2-25) and Eq. (13.2-28), along with Eq. (13.2-29), the definition of ϕ_{hf}.

The constraint for negative ϕ_{hf} (freezing) arises from the condition that more cladding cannot freeze then exists in the molten state. Expressed mathematically, the limit is

$$\phi_{hf} \geq -\frac{A_c \rho_c \lambda_c}{\Delta t P_r} \tag{13.3-41}$$

13.3.2.6 Intact and Moving Cladding

The molten cladding energy equation for this case is identical to that for the case of moving cladding on intact and refrozen cladding. The molten cladding temperature change is therefore given by Eq. (13.3-36), which is also displayed below

$$\Delta T_{c,j}^n = -\frac{\left[\xi_2 \left(T_m - T_c\right) - \gamma_c\right] \Delta t}{1 + \xi_2 \Delta t}\Bigg|_j^n \tag{13.3-42}$$

The quantities ϕ_1, ϕ_2, and ϕ_{trial} are computed from Eq. (13.2-41) to Eq. (13.2-43). Then ϕ_{hf} and ϕ_r are computed according to Eq. (13.2-44), which we repeat below

$$\phi_{hf} = 0, \quad \phi_r = \phi_2 \quad \text{for } \phi_{trial} \geq 0, \tag{13.3-43}$$

$$\phi_{\text{hf}} = \phi_{\text{trial}}, \quad \phi_{\text{r}} = \phi_1 \quad \text{for } \phi_{\text{trial}} < 0. \tag{13.3-44}$$

The case of $\phi_{\text{trial}} \geq 0$ corresponds to melting of the initial cladding; the heat of fusion is included in ϕ_{r} which is the negative of the heat flux to the pin surface.

For $\phi_{\text{trial}} \leq 0$, we obtain a negative ϕ_{hf} (freezing) which will result in the appearance of refrozen cladding during the current time step. The constraint given by Eq. (13.3-41) is checked and, if not satisfied, then (i) ϕ_{hf} is set equal to the limit given by Eq. (13.3-41) and (ii) a new ϕ_{r} is computed by

$$\phi_{\text{r}} = \phi_2 + \phi_{\text{hf}} \tag{13.3-45}$$

13.3.2.7 Heat Loss to Structure

At the onset of cladding motion, the structure is still relatively cool and consequently constitutes a potentially significant heat sink for the refreezing of cladding, particularly in small experimental subassemblies. The local heat absorbed by the structure (per unit length and unit time) is given by

$$Q^n_{\text{s,j}} = \frac{\theta P_{\text{w}} \left(T_{\text{m}} - T_{\text{w}}\right)}{\frac{\Delta r_{\text{w}}}{k} + \frac{1}{h}} \tag{13.3-46}$$

where

θ = multiplier on heat loss to structure (usually = 1)

P_{w} = heated perimeter of the structure

T_{w} = structure temperature

Δr_{w} = half-thickness of structure.

The present CLAP model allows for the indirect transfer of heat form the moving cladding to the structure by way of the frozen cladding. An adjustment to the refrozen clad temperature ΔT_{s} is defined by

$$T^{n+1}_{\text{s,j}} = \hat{T}^{n+1}_{\text{s,j}} + \Delta\hat{T}^n_{\text{s,j}} \tag{13.3-47}$$

where $\hat{T}^{n+1}_{\text{s,j}}$ is the unadjusted refrozen cladding temperature.

Normally, the temperature adjustment would be given by

$$\Delta\hat{T}^n_{\text{s,j}} = Q_{\text{s}} \frac{\Delta t}{A_{\text{s}}\rho_{\text{s}}c_{\text{ps}}}\bigg|^n_{\text{j}} \tag{13.3-48}$$

However, this formulation would be unstable in the case of vanishing A_{s} or large Δt. To achieve stability, we slightly alter the physics; in particular, T_{m} is replaced in Eq. (13.3-46) by $T^{n+1}_{\text{s,j}}$. This adjustment in T_{s} is made only for segments where moving cladding is present; and, for these segments, we would expect the frozen cladding temperature to be close to the melting temperature, and hence the error in this approximation would be small. With this modification the adjustment is given by

$$\Delta\hat{T}^n_{\text{s,j}} = \frac{\xi_{\text{w}}\Delta t^n \left(\Delta\hat{T}^{n+1}_{\text{s,j}} - T_{\text{w}}\right)}{1 + \epsilon_{\text{w}}\Delta t^n} \tag{13.3-49}$$

where

$$\xi_{\text{w}} = \frac{\theta P_{\text{w}}}{A_{\text{s}}\rho_{\text{s}}C_{\text{ps}}\left(\frac{\Delta r_{\text{w}}}{k} + \frac{1}{h}\right)}\bigg|^n_{\text{j}} \tag{13.3-50}$$

13.3.3 Refrozen Steel Area Calculation

The refrozen cladding mass conservation Eq. (13.2-26) with the source term given by Eq. (13.2-30) is rewritten in the form

$$\rho_{\mathrm{s}} \frac{\partial \mathrm{A_s}}{\partial \mathrm{t}} = -\frac{P_{\mathrm{r}} \phi_{\mathrm{hf}}}{\lambda} - A_{\mathrm{s}} \frac{\partial \rho_{\mathrm{s}}}{\partial \mathrm{t}}. \tag{13.3-51}$$

The finite difference form of this equation is

$$A_{\mathrm{s,j}}^{n+1} = A_{\mathrm{s,j}}^{n} - \frac{\left[\left. \frac{\phi_{\mathrm{hf}} P_{\mathrm{r}}}{\lambda} \Delta t \right|_{\mathrm{j}}^{n} + A_{\mathrm{s,j}}^{n} \left(\rho_{\mathrm{s,j}}^{n+1} - \rho_{\mathrm{s,j}}^{n} \right) \right]}{\rho_{\mathrm{s,j}}^{n+1}} \tag{13.3-52}$$

After the density $\rho_{\mathrm{s,j}}^{n+1}$ is evaluated from $T_{\mathrm{s,j}}^{n+1}$ using Eq. (13.2-46), then the area $A_{\mathrm{s,j}}^{n+1}$ is computed using Eq. (13.3-52). The thickness of the refrozen steel layer is computed from its area by assuming that the refrozen steel forms an annular ring around the intact cladding.

13.3.4 Moving Cladding Area Calculation

The continuity Eq. (13.2-14) and source term Eq. (13.2-30) are rewritten as

$$\rho_{\mathrm{c}} \frac{\partial \mathrm{A_c}}{\partial \mathrm{t}} = -A_{\mathrm{c}} \frac{\partial \rho_{\mathrm{c}}}{\partial \mathrm{t}} - C_{\mathrm{a}} + \frac{P_{\mathrm{r}} \varphi_{\mathrm{hf}}}{\lambda} \tag{13.3-53}$$

where

$$C_{\mathrm{a}} = \frac{\partial}{\partial \mathrm{z}} \left(\rho_{\mathrm{c}} A_{\mathrm{c}} v_{\mathrm{c}} \right) \tag{13.3-54}$$

The finite difference form of this equation is given by

$$A_{\mathrm{c,j}}^{n+1} = A_{\mathrm{c,j}}^{n} + \frac{\left[-A_{\mathrm{c,j}}^{n} \left(\rho_{\mathrm{c,j}}^{n+1} - \rho_{\mathrm{c,j}}^{n} \right) - C_{\mathrm{a}} \Delta t_{\mathrm{n}} + \left. \frac{P_{\mathrm{r}} \phi_{\mathrm{hf}} \Delta t}{\lambda} \right|_{\mathrm{j}}^{n} \right]}{\rho_{\mathrm{c,j}}^{n+1}} + A_{\mathrm{mod,j}}^{n} \tag{13.3-55}$$

where A_{mod} is a correction term, defined later.

The solution method uses donor-cell differencing to evaluate the convective term C_{a} in terms of the nodal fluxes w_{j} (see Eq. (13.3-16)) and nodal elevation (see Eq. (13.3-15)). Normally $A_{\mathrm{mod}} = 0$ and C_{a} is computed as follows

$$C_{\mathrm{a}} = \frac{w_{\mathrm{j}}^{n} - w_{\mathrm{j-1}}^{n}}{z_{\mathrm{m,j}} - z_{\mathrm{m,j-1}}} \quad \text{for } w_{\mathrm{m,j}} \geq 0 \tag{13.3-56}$$

$$C_{\mathrm{a}} = \frac{w_{\mathrm{j+1}}^{n} - w_{\mathrm{j}}^{n}}{z_{\mathrm{m,j+1}} - z_{\mathrm{m,j}}} \quad \text{for } w_{\mathrm{m,j}} < 0 \tag{13.3-57}$$

Modifications to this formulation are made (i) to prevent cladding transfer outside of the fuel and blanket region and (ii) to inhibit cladding transfer to flooded or blocked nodes; the modifications consist simply of setting the appropriate values of w_{j} equal to zero.

Additionally, adjustments must be made for the case where the molten cladding area exceeds the available area A_{max}, defined by

$$A_{\mathrm{max,j}}^{n+1} = A_{\mathrm{f,j}} - A_{\mathrm{i,j}}^{n+1} - A_{\mathrm{s,j}}^{n+1} \tag{13.3-58}$$

where

A_{f} = total area for cladding allowed by the fuel

A_{i} = area of the intact cladding (if any).

For the case where the computed $A_{\mathrm{c,j}}^{n+1}$ exceeds $A_{\max}$, the following steps are taken: (i) the condition is flagged by setting $\mathrm{NFULL_j} = 1$ (initial value is 0), (ii) the molten cladding area $A_{\mathrm{c,j}}^{n+1}$ is set equal to $A_{\max}$, and (iii) the computed area for the donor segment is later adjusted consistent with step (ii).

To accomplish step (iii), we first estimate the volume of cladding convected into segment j, denoted by Γ_{j}, from continuity considerations:

$$\Gamma_{\mathrm{j}} \cong \left[A_{\mathrm{max,j}}^{n+1} - A_{\mathrm{c,j}}^{n} - \frac{P_{\mathrm{r}} \frac{\phi_{\mathrm{hf}} \Delta t}{\lambda}\Big|_{\mathrm{j}}^{n}}{\rho_{\mathrm{c,j}}^{n+1}} \right] (z_{\mathrm{j+1}} z_{\mathrm{j}}) \tag{13.3-59}$$

Rather than adjusting C_{a} for the donor cell, we instead set w_{j} (or $w_{\mathrm{j+1}}$) equal to zero and compute a correction A_{mod}, which is the area change (negative) due to the volume of fluid Γ_{j} transferred to the flooded cell. For example, for the case of the donor cell below the flooded cell we have

$$w_{\mathrm{j}} = 0, A_{\mathrm{mod,j-1}}^{n} = -\frac{\Gamma_{\mathrm{j}}}{(z_{\mathrm{j}} - z_{\mathrm{j-1}})} \quad \text{for } \mathrm{NFULL_j} = 1, w_{\mathrm{m,j-1}} \geq 0. \tag{13.3-60}$$

Similarly, for the case of the donor cell above the flooded cell we have

$$w_{\mathrm{j+1}} = 0, A_{\mathrm{mod,j+1}}^{n} = -\frac{\Gamma_{\mathrm{j}}}{(z_{\mathrm{j+2}} - z_{\mathrm{j+1}})}, \quad \text{for } \mathrm{NFULL_j} = 1, w_{\mathrm{m,j+1}} < 0. \tag{13.3-61}$$

13.3.5 Reactivity Calculation

The reactivity change due to cladding relocation is computed by

$$\Delta K = \sum_{\mathrm{j}} \left(m_{\mathrm{j}}^{n} - m_{\mathrm{j}}^{o} \right) \cdot W_{\mathrm{j}} \tag{13.3-62}$$

where

m_{j}^{n} = mass of cladding in the j-th segment

m_{j}^{o} = original mass of cladding in the j-th segment

W_{j} = cladding reactivity worth distribution.

In the molten region, the mass is computed by

$$m_{\mathrm{j}}^{n} = A_{\mathrm{c}} \rho_{\mathrm{c}} \Delta z \Big|_{\mathrm{j}}^{n} \tag{13.3-63}$$

$$\text{where } \Delta z_{\mathrm{j}} = z_{\mathrm{j+1}} - z_{\mathrm{j}} \tag{13.3-64}$$

Outside the molten region the mass is evaluated using

$$m_{\mathrm{j}}^{n} = m_{\mathrm{j}}^{0} + A_{\mathrm{s}} \rho_{\mathrm{s}} \Delta z \Big|_{\mathrm{j}}^{n} + A_{\mathrm{c}} \rho_{\mathrm{c}} \Delta z \Big|_{\mathrm{j}}^{n}. \tag{13.3-65}$$

Due to roundoff and other possible small errors, the total mass of cladding may not be absolutely conserved which, if left uncorrected, would create anomalous reactivity effects. To insure mass conservation, the computed mass distribution is renormalized by the following steps:

$$M^n = \sum_{\mathrm{j}} m_{\mathrm{j}}^n, \tag{13.3-66}$$

$$M^o = \sum_{\mathrm{j}} m_{\mathrm{j}}^o, \tag{13.3-67}$$

$$m_{\mathrm{j}}^n \cdot \frac{M^o}{M^n} \rightarrow m_{\mathrm{j}}^n \tag{13.3-68}$$

where the operation A → B indicates that A is substituted for B.

13.3.6 Moving Cladding Velocity Calculation

The cladding momentum equation is solved in a separate subroutine (TSCLD2) that is called after the coolant dynamics equations have been solved. The terms $\frac{\partial \mathrm{p}}{\partial \mathrm{z}}$ and F_{v} in the cladding momentum equation are evaluated at time $t_{\mathrm{n}} + 1/2\Delta t^n$ (by averaging values at t^n and t^{n+1}) to achieve maximum accuracy with respect to sodium-vapor-induced cladding accelerations.

The pin friction term, $\left(\frac{\partial \mathrm{p}}{\partial \mathrm{z}}\right)_{\mathrm{fr}}$, is evaluated at the end of the time step to obtain maximum stability under conditions of strongly accelerating thin cladding films. Substituting

$$v_{\mathrm{c,j}}^{n+1} = v_{\mathrm{c,j}}^n + \Delta v_{\mathrm{c,j}}^n \tag{13.3-69}$$

into Eq. (13.2-17) and Eq. (13.2-18) and retaining only linear terms, we obtain an equation of the form

$$F_{\mathrm{p,j}}^{n+1} = y_{1,\mathrm{j}}^n + y_{2,\mathrm{j}}^n \Delta v_{\mathrm{c,j}}^n \tag{13.3-70}$$

where

$$y_{1,\mathrm{j}}^{n+1} = \left.\frac{32\mu_{\mathrm{c}} v_{\mathrm{c}}}{D_{\mathrm{c}}^2}\right|_{\mathrm{j}}^n \quad \text{for } \mathrm{Re}_{\mathrm{j}}^n < (\mathrm{Re})_{\mathrm{break}} \tag{13.3-71}$$

$$y_{2,\mathrm{j}}^n = \left.\frac{32\mu_{\mathrm{c}}}{D_{\mathrm{c}}^2}\right|_{\mathrm{j}}^n \quad \text{for } \mathrm{Re}_{\mathrm{j}}^n < (\mathrm{Re})_{\mathrm{break}} \tag{13.3-72}$$

$$y_{1,\mathrm{j}}^n = \left.\frac{b_{\mathrm{f}} \rho_{\mathrm{c}} v_{\mathrm{c}} \left|v_{\mathrm{c}}\right|}{2D_{\mathrm{c}}}\right|_{\mathrm{j}}^n \quad \text{for } \mathrm{Re}_{\mathrm{j}}^n \geq (\mathrm{Re})_{\mathrm{break}} \tag{13.3-73}$$

$$y_{2,\mathrm{j}}^n = \left.\frac{b_{\mathrm{f}} \rho_{\mathrm{c}} \left|v_{\mathrm{c}}\right|}{D_{\mathrm{c}}}\right|_{\mathrm{j}}^n \quad \text{for } \mathrm{Re}_{\mathrm{j}}^n \geq (\mathrm{Re})_{\mathrm{break}} \tag{13.3-74}$$

The parameter C_{m} is used to represent the momentum convective term

$$C_{\mathrm{m,j}} = v_{\mathrm{c}} \left.\frac{\partial \mathrm{v_c}}{\partial \mathrm{z}}\right|_{\mathrm{j}} \tag{13.3-75}$$

This parameter is evaluated by alternative formulae depending upon whether the segment is filled with cladding. For NFULL $(j) = 1$, indicating a filled segment, the term is estimated by the formula

$$C_{\mathrm{m,j}}^n = v_{\mathrm{c,j}}^n \left.\frac{\partial \mathrm{v_c}}{\partial \mathrm{z}}\right|_{\mathrm{j}}^{n+1/2} \tag{13.3-76}$$

where the derivative $\frac{\partial \mathrm{v_c}}{\partial \mathrm{z}}$ is obtained by rearranging the continuity equation:

$$\frac{\partial \mathrm{v_c}}{\partial \mathrm{z}} = \frac{1}{\rho_\mathrm{c} A_\mathrm{c}} \left[\frac{\phi_\mathrm{hf} P_\mathrm{r}}{\lambda} - \frac{\partial \left(\rho_\mathrm{c} A_\mathrm{c} \right)}{\partial \mathrm{t}} - v_\mathrm{c} \frac{(\rho_\mathrm{c} A_\mathrm{c})}{\partial \mathrm{z}} \right] \tag{13.3-77}$$

and evaluating at time $t^n + 1/2\Delta t^n$. Two alternative equations are obtained, one for each flow direction, as given by the following:

$$\left. \frac{\partial \mathrm{v_c}}{\partial \mathrm{z_j}} \right|^{n+1/2} = \frac{1}{(\rho_{\mathrm{c,j}} A_{\mathrm{c,j}})^{n+1/2}} \left[\left. \frac{\phi_{\mathrm{h_f}} P_\mathrm{r}}{\lambda_\mathrm{c}} \right|_\mathrm{j}^n - \rho_{\mathrm{c,j}}^n \frac{A_{\mathrm{c,j}}^{n+1} - A_{\mathrm{c,j}}^n}{\Delta t^n} \right.$$
$$\left. - A_{\mathrm{c,j}}^n \frac{\rho_{\mathrm{c,j}}^{n+1} - \rho_{\mathrm{c,j}}^n}{\Delta t^n} - v_{\mathrm{c,j}}^n \frac{(\rho_{\mathrm{c,j+1}} A_{\mathrm{c,j+1}})^{n+1/2} - (\rho_{\mathrm{c,j}} A_{\mathrm{c,j}})^{n+1/2}}{z_{\mathrm{j+1}} - z_\mathrm{j}} \right] \tag{13.3-78}$$

for $w_{\mathrm{m,j}}^n < 0$

$$\left. \frac{\partial \mathrm{v_c}}{\partial \mathrm{z_j}} \right|^{n+1/2} = \frac{1}{(\rho_{\mathrm{c,j}} A_{\mathrm{c,j}})^{n+1/2}} \left[\left. \frac{\phi_{\mathrm{h_f}} P_\mathrm{r}}{\lambda_\mathrm{c}} \right|_\mathrm{j}^n - \rho_{\mathrm{c,j}}^n \frac{A_{\mathrm{c,j}}^{n+1} - A_{\mathrm{c,j}}^n}{\Delta t^n} \right.$$
$$\left. - A_{\mathrm{c,j}}^n \frac{\rho_{\mathrm{c,j}}^{n+1} - \rho_{\mathrm{c,j}}^n}{\Delta t^n} - v_{\mathrm{c,j}}^n \frac{(\rho_{\mathrm{c,j+1}} A_{\mathrm{c,j+1}})^{n+1/2} - (\rho_{\mathrm{c,j}} A_{\mathrm{c,j}})^{n+1/2}}{z_{\mathrm{j+1}} - z_\mathrm{j}} \right] \tag{13.3-79}$$

for $w_{\mathrm{m,j}}^n < 0$

where quantities on the right-hand side of Eq. (13.3-78) and Eq. (13.3-79) with the superscript $n + 1/2$ are evaluated at time $t^n + \Delta t^{n/2}$ as follows

$$(\ )^{n+1/2} = 1/2 \left[(\ )^n + (\ )^{n+1} \right] \tag{13.3-80}$$

For $\mathrm{NFULL_j} = 0$, indicating an unblocked segment, we use the identity

$$v_\mathrm{c} \frac{\partial \mathrm{v_c}}{\partial \mathrm{z}} = 1/2 \frac{\partial \mathrm{v_c^2}}{\partial \mathrm{z}} \tag{13.3-81}$$

and evaluate the momentum convection term as follows

$$C_{\mathrm{m,j}} = \frac{\frac{1}{2} \left[\left(v_{\mathrm{c,j+1}}^n \right)^2 - \left(v_{\mathrm{c,j}}^n \right)^2 \right]}{(z_{\mathrm{m,j+1}} - z_{\mathrm{m,j}})} \quad \text{for } w_{\mathrm{m,j}}^n < 0 \tag{13.3-82}$$

$$C_{\mathrm{m,j}} = \frac{\frac{1}{2} \left[\left(v_{\mathrm{c,j}}^n \right)^2 - \left(v_{\mathrm{c,j-1}}^n \right)^2 \right]}{(z_{\mathrm{m,j}} - z_{\mathrm{m,j-1}})} \quad \text{for } w_{\mathrm{m,j}}^n \geq 0 \tag{13.3-83}$$

The discretized version of he moving-cladding momentum Eq. (13.2-24), using the linear approximation for F_p Eq. (13.3-70), and the symbolic representation for the momentum convective term Eq. (13.3-75), is given by

$$(\rho_{\mathrm{c,j}})^{n+1/2} \left[\frac{\Delta v_{\mathrm{c,j}}^n}{\Delta t^n} + C_{\mathrm{m,j}} + g \right] = - \left. \frac{\partial \mathrm{P}}{\partial \mathrm{z}} \right|_\mathrm{j}^{n+1/2} - \left. \frac{A_\mathrm{v}}{A_\mathrm{c}} F_\mathrm{v} \right|_\mathrm{j}^{n+1/2} -$$
$$y_{1,\mathrm{j}}^n - y_{2,\mathrm{j}}^n \Delta v_{\mathrm{c,j}}^n - \begin{cases} 0 & \text{if freezing} \quad \text{(13.3-78a)} \\ \dfrac{P_{\mathrm{r,j}}^n \phi_{\mathrm{hf,j}}^n \left(v_{\mathrm{c,j}}^n + \Delta v_{\mathrm{c,j}}^n \right)}{A_{\mathrm{c,j}}^{n+1/2} \lambda_{\mathrm{c,j}}^n} & \text{if melting} \quad \text{(13.3-78b)} \end{cases} \tag{13.3-84}$$

Solving this equation for Δv_c, one obtains the form used in the CLAP program.

13.4 Input-Output Description

A listing of CLAP-related input variables is given in Table 13.4.1 as an aid in preparing or modifying SAS input decks. Additionally, the input items on this list should be rechecked when problems are encountered in the running of SAS cases with cladding motion. This list contains all input items used directly by the following CLAP subroutines: TSCLD1, TSCLD2, SODFRC and DENSIT and, in addition, input items used to initiate CLAP and to calculate reactivity effects due to cladding motion (in subroutine FEEDBK).

The CLAP output appears with both the main full printout and the boiling printout. A sample output is shown in Figure 13.4.1. The output labels are generally self-explanatory. The Fortran variable and units corresponding to each output item are given in Table 13.4.2.

Table 13.4.1: Listing of CLAP-Related Input Parameters

Symbol	Reference Eq. No.	Reference	Block	Location	Recommended Values
T_{ref}	Eq. (13.2-46)	TR	13	419	1700 K
T_{m}	—	TESOL*	13	810-812	1700 K [13-7]
λ^o	Eq. (13.2-51)	UEMELT	13	816-818	2.703×10^5 J/kg [13-7]
λ^o	Eq. (13.2-51)	UEMELT	13	816-818	2.703×10^5 J/kg [13-7]
C_{ps}	Eq. (13.2-49)	CE	13	1070-1072	690.1 J/kg K [13-7]
		ICLADV	51	17	—
		ICLADB	51	85	1
		IDCLGO	51	125	0
		IDCLSP	51	126	0
		IDCLDE	51	127	0
W_{j}	Eq. (13.3-62)	CLADRA	62	160-183	—
ρ_{s}^o	Eq. (13.2-46)	DENSS	63	35	7.256×10^3 kg/m^{3†} [13-7]
C_1, C_2, C_3	Eq. (13.2-33)	C1, C2, C3	64	3-5	Eq. (13.2-33) [13-8]
AFRV,BFRV	Eq. (13.2-5)	AFRV, BFRV	64	168, 169	0.316, -0.25
ρ_{c}^o	Eq. (13.2-47)	DENS	65	3	6.98×10^3 kg/m^{3†} [13-7]
β	Eq. (13.2-46)	AE	65	4	2.3×10^{-5}/K† [13-7]
C	Eq. (13.2-47)	RHOCD	65	5	8.3×10^{-5}/K† [13-7]
μ_{m}	Eq. (13.2-21)	VISMC	65	6	6.42×10^{-3} Pa-s [13-7]
μ_{T}	Eq. (13.2-22)	VISTR	65	7	1×10^4 Pa-s [13-3]
μ_{s}	Eq. (13.2-23)	VISSC	65	8	1×10^4 Pa-s [13-3]
q	Eq. (13.2-22)	XVISC	65	9	0.5
b_{f}	Eq. (13.2-19)	CLADFR	65	10	0.02 [13-3]

continues on next page

Table 13.4.1 – continued from previous page

Symbol	Reference Eq. No.	Reference	Block	Location	Recommended Values
$(\mathrm{Re})_{\mathrm{break}}$	Eq. (13.2-18)	REBRK	65	11	2100. [13-3]
ε	Eq. (13.2-10)	GMULTF	65	12	75.0 [13-5]
α_{crit}	Eq. (13.2-10)	ALPHCR	65	13	0.0 [13-5]
$(\mathrm{fps})_{\mathrm{o}}$	Eq. (13.2-12)	FPSO	65	14	0.3s
x	Eq. (13.2-12)	EXPFPS	65	15	3.0
θ	Eq. (13.3-46)	CLSTHR	65	17	1.0
a	Eq. (13.2-21)	AVISC	65	20	5492 K [13-7]
c_{pc}	Eq. (13.2-50)	CPC	65	21	621.7 J/kg K† [13-7]

*Orginally TME, but TME is now set equal to the solidus temperature TESOL.

†Evaluated at the melting temperature of 1700 K.

Table 13.4.2: Summary of CLAP Output Items

Item	Symbol	FORTRAN Variable	Units
CLAP Time Step Count	n	ICOUNT	—
Axial Coolant Mode	j	J	—
Main Axial (Nodal) Elevation	$z_{\mathrm{m,j}}$	ZFM	m
Refrozen Cladding Thickness	$2\Delta r_{\mathrm{s}}$	RCLAD1	m
Moving Cladding Thickness	$2\Delta r_{\mathrm{c}}$	TCLAD1	m
Intact Cladding Temperature	T_{i}	TCLAV	K
Refrozen Cladding Temperature	T_{s}	TEMFS1	K
Moving Cladding Temperature	T_{c}	TEMMC1	K
Moving Cladding Velocity	V_{c}	CVEL1	m/s
Coolant Flow Area	A_{v}	AREAC1	m^2
Internal Flag (=1 in molten zone, =0 otherwise)		LVEL	—

13.5 References

13-1. W. R. Bohl and T. J. Heames, "A Cladding Motion Model for LMFBR Loss-of-Flow Accident Analysis," *Trans. Am. Nucl. Soc.*, vol. 17, p. 358, 1973.

13-2. W. R. Bohl and C. H. Bowers, Reactor Analysis and Safety Division, Argonne National Laboratory, personal communication, December 1975.

13-3. W. R. Bohl and T. J. Heames, Unpublished information, Argonne National Laboratory, 1974.

13-4. H. K. Fauske, et al., Unpublished information, Argonne National Laboratory, 1974.

13-5. G. B. Wallis, *One-Dimensional Two-Phase Flow*, McGraw-Hill, New York, 1969.

13-6. T. G. Theofanous, M. DiMonte, and P. D. Patel, "Incoherency Effects in Clad Relocation Dynamics for LMFBR CDA Analyses," *Nucl. Eng. Des.*, vol. 36, p. 59, 1976.

CHANNEL 1 SAS9A 0.0 R5CASE10
SAMPLE OUTPUT FOR SAS DOCUMENT

CLAP OUTPUT, CHANNEL NUMBER 1 TIME = 16.91526 ICOUNT = 892

AXIAL COOLANT NODE	MEAN AXIAL ELEVATION	REFROZEN CLAD THICKNESS	MOVING CLAD THICKNESS	* * * TEMPERATURES * * * INTACT CLAD	REFROZEN CLAD	MOVING CLAD	MOVING CLAD VELOCITY	COOLANT FLOW AREA	TOTAL CLAD MASS	INTERNAL FLAGS
21	1.2200+00	5.3400-04	0.0	1230.0	1222.9	1644.1	0.0	9.742D-06	6.861D-03	0
20	1.1720+00	5.2460-04	0.0	1534.4	1547.7	1644.1	2.251D-12	1.126D-05	6.240D-03	0
19	1.1240+00	9.2600-04	0.0	1267.3	1248.6	1644.2	1.996D-11	2.063D-06	9.640D-03	0
18	1.0900+00	8.4270-04	0.0	1310.7	1281.6	1644.2	0.0	3.912D-06	3.771D-03	0
17	1.0410+00	0.0	1.4150-04	0.0	0.0	2121.4	-2.142D-01	2.713D-05	1.147D-03	1
16	9.6520-01	0.0	1.5060-04	0.0	0.0	2184.2	-2.417D-01	2.698D-05	1.216D-03	1
15	8.8960-01	0.0	1.3910-04	0.0	0.0	2235.8	-2.257D-01	2.717D-05	1.114D-03	1
14	8.1280-01	0.0	1.5930-04	0.0	0.0	2246.8	-3.034D-01	2.684D-05	1.279D-03	1
13	7.3600-01	0.0	1.9380-04	0.0	0.0	2171.2	-3.552D-01	2.626D-05	1.579D-03	1
12	6.6040-01	0.0	1.7680-04	0.0	0.0	2050.7	-3.193D-01	2.655D-05	1.455D-03	1
11	5.8420-01	0.0	1.7580-04	0.0	0.0	1925.9	-3.437D-01	2.655D-05	1.465D-03	1
10	5.0800-01	3.593D-05	2.189D-06	1547.4	1629.4	1644.2	-1.727D-03	2.165D-05	4.338D-03	0
9	4.3180-01	0.0	0.0	1372.6	0.0	0.0	0.0	2.175D-05	3.945D-03	0
8	3.5560-01	0.0	0.0	1245.8	0.0	0.0	0.0	2.175D-05	3.945D-03	0
7	2.7940-01	0.0	0.0	1224.1	0.0	0.0	0.0	2.175D-05	3.945D-03	0
6	2.0320-01	0.0	0.0	1226.4	0.0	0.0	0.0	2.175D-05	3.945D-03	0
5	1.5490-01	0.0	0.0	1190.2	0.0	0.0	0.0	2.175D-05	1.052D-03	0
4	1.2060-01	0.0	0.0	1167.8	0.0	0.0	0.0	2.175D-05	2.498D-03	0
3	7.2390-02	0.0	0.0	956.6	0.0	0.0	0.0	2.175D-05	2.498D-03	0
2	2.4130-02	0.0	0.0	658.7	0.0	0.0	0.0	2.175D-05	2.498D-03	0

Figure 13.4.1: Sample CLAP Output

13-7. L. Leibowitz, et al., Unpublished information, Argonne National Laboratory, 1976.

13-8. M. W. Maresco and O. E. Dwyer, “Heat Transfer to Mercury Flowing In-Line Through A Bundle of Circular Rods,” *Trans. A.S.M.E.-J. Heat Transfer*, vol. 86, p. 180, May 1964.

13.6 Nomenclature

Symbol	Definition	Units
A_c	Moving cladding cross-sectional area	m^2
AFRV	Input constant in single-phase friction factor formula, Eq. (13.2-5)	
A_f	Total area allowed for cladding by the fuel	m^2
A_{max}	Available area for molten cladding	m^2
A_s	Refrozen steel cross-sectional area	m^2
A_v	Vapor flow area	m^2
a	Constant in viscosity Eq. (13.2-21)	K
BFRV	Input constant, see AFRV	–
b_f	Molten cladding/pin turbulent friction factor, Eq. (13.2-19)	–
C	Liquid-steel volumetric coefficient of thermal expansion, Eq. (13.2-47)	K^{-1}
C_a	Mass convection term	kg/m-s
C_f	Terms in the outer-fuel-node energy Eq. (13.3-21)	W/m
C_m	Momentum convection term	m/s^2
C_v	Energy convection term, Eq. (13.3-9)	W/m
C_1, C_2, C_3	Input constants in the correlation of liquid metal heat transfer,	
c_f	Eq. (13.2-33)	–
c_{pc}	Coefficient of friction with fuel pin, Eq. (13.2-17)	–
c_{pf}	Molten cladding specific heat capacity	J/kg-K

continues on next page

Table 13.6.1 – continued from previous page

Symbol	Definition	Units
c_{ps}	Fuel specific heat capacity	J/kg-K
D_c	Solid steel specific heat capacity	J/kg-K
D_c	Molten cladding hydraulic diameter	m
D_h	Hydraulic diameter for bare fuel or fuel pin	m
D_v	Hydraulic diameter for the vapor	m
e_c	Moving cladding internal energy	J/kg
e_c^o	Constant, Eq. (13.2-50)	J/kg
e_s	Refrozen steel internal energy	J/kg
F_p	Pin/molten-cladding friction force per unit volume of molten cladding	N/m^3
F_v	Cladding/vapor interfacial force per unit volume of channel	N/m^3
f	Melt fraction	–
fps	Full-power seconds from initial cladding motion	s
$(\text{fps})_0$	Constant in incoherence factor on friction, Eq. (13.2-12)	s
f_{sf}	Single-phase friction factor for vapor	–
g	Gravitational constant	m/s^2
h	Coefficient of heat transfer to the molten cladding from the solid interface	W/m^2-K
I	Incoherence multiplier on friction	–
j	Index for axial segment	–
ΔK	Reactivity change due to cladding relocation	δk/k
k_c	Molten cladding thermal conductivity	W/m-K
k_f	Fuel thermal conductivity	W/m-K
M	Friction multiplier due to flooding (M is also the total steel mass in channel)	–
m_j	Mass of cladding in segment j	kg
m_j^o	Initial mass of cladding in segment j	kg
$\dot{m}_c$	Mass rate of cladding melting per unit length of channel	kg/m-s
$\dot{m}_v$	Rate of vapor generation per unit length of channel	kg/m-s
n	Index for time step	–
P_e	Outer perimeter of intact cladding	m
P_r	Perimeter of the cladding solid/liquid interface	m
P/D	Pitch-to-diameter ratio for fuel pins	–
$\frac{\partial p}{\partial z}$	Channel axial pressure gradient	Pa/m
Q_j	Cumulative pin segment heat loss from beginning of heat transfer time step	J
Q_{NT}	Volumetric heat generation in outer fuel segment	W/m^3
q	Input constant, Eq. (13.2-22)	–
Re	Molten cladding Reynolds number, Eq. (13.2-20)	–
$(\text{Re})_{break}$	Turbulent transition Reynolds number, Eq. (13.2-18)	–
$(\text{Re})_v$	Reynolds number for vapor	–
r	Radius (from fuel pin axis)	m
r_{NR}, r_{NT}	Fuel radii defined in Figure 13.3.1	m
Δr_c	Half-thickness of molten cladding layer	m
Δr_i	Half-thickness of the intact cladding	m
Δr_s	Half-thickness of the refrozen cladding	m
Δr_w	Half-thickness of the structure	m
T_c	Moving cladding temperature	K
T_f	Fuel surface temperature	K
T_i	Intact cladding temperature	K
T_m	Cladding melting temperature	K
T_{ref}	Reference temperature in density Eq. (13.2-46)	K
T_s	Refrozen cladding temperature	K

continues on next page

Table 13.6.1 – continued from previous page

Symbol	Definition	Units
T_w	Structure temperature	K
t	Time	s
t^*	Time at beginning of current heat-transfer time step	s
Δt	CLAP (coolant) time step	s
Δt^*	Heat-transfer time step	s
v_c	Moving cladding velocity	m/s
v_{flood}	Flooding velocity	m/s
W_j	Cladding reactivity worth distribution	∂k/k-kg
w	Vapor mass flowrate	kg/s
w_j	Segment midpoint mass flow, Eq. (13.3-16)	kg/s
w_j^*	Segment boundary mass flow, Eq. (13.3-10)	kg/s
$w_{m,j}$	Segment mean mass flow, Eq. (13.3-11)	kg/s
x	Constant in incoherence factor on friction, Eq. (13.2-12)	–
y_1, y_2	Constants in linearized pin friction equation	N/m^3
z	Elevation	m
z_j	Segment boundary elevation	m
$z_{m,j}$	Nodal elevation, Eq. (13.3-15)	m
Δz_j	Segment length	m
α	Vapor fraction based on area available for molten steel and vapor	–
α_{crit}	Input constant in two-phase multiplier, Eq. (13.2-10)	–
β	Steel (solid) coefficient of linear thermal expansion, Eq. (13.2-46)	K^{-1}
Γ	Factor in correction of cladding area for overfilled segments	m^3
γ_c	Computer coefficient, Eq. (13.3-26)	
γ_f	Computed coefficient, Eq. (13.3-23)	
ε	Input constant in two-phase multiplier, Eq. (13.2-10)	–
θ	Multiplier on heat loss to structure (usually = 1)	–
λ	Effective heat-of-fusion, Eq. (13.2-31)	J/kg
λ_o	Thermodynamic heat-of-fusion	J/kg
μ_c	Moving cladding viscosity	Pa-s
μ_m	Cladding viscosity at the liquidus temperature	Pa-s
μ_s	Solid cladding pseudo-viscosity, Eq. (13.2-23)	Pa-s
μ_t	Cladding viscosity at the solidus temperature	Pa-s
μ_v	Vapor viscosity	Pa-s
ξ_f, ξ_w	Computed coefficients	–
ξ_1, ξ_2, ξ_3	Computed coefficients	–
ρ_c	Molten cladding density	kg/m^3
ρ_c°	Density of cladding at the liquidus temperature	kg/m^3
ρ_f	Fuel density	kg/m^3
ρ_s	Refrozen steel density	kg/m^3
ρ_s°	Solid cladding density at the reference temperature	kg/m^3
ρ_v	Vapor density	kg/m^3
ϕ	Sensible heat flux from refrozen cladding to the molten interface	W/m^2
ϕ_c	Flux of sensible heat into the moving cladding layer	W/m^2
ϕ_{hf}	Fusion heat flux, Eq. (13.2-29)	W/m^2
ϕ_r	Heat flux at interface of intact and refrozen cladding	W/m^2
ϕ_{trial}	Trail heat flux, Eq. (13.2-43)	W/m^2
ϕ_1	Heat flux, Eq. (13.2-41)	W/m^2
ϕ_2	Heat flux, Eq. (13.2-42)	W/m^2

continues on next page

Table 13.6.1 – continued from previous page

Symbol	Definition	Units
$\overline{\psi}$	Mean ratio of thermal-to-momentum eddy diffusivities	–

CHAPTER

FOURTEEN

PLUTO2: NON-VOIDED CHANNEL FUEL MOTION ANALYSIS

14.1 Introduction and Overview

14.1.1 Historical Background and Description of the Physical Model

The PLUTO2 module calculates the post-failure fuel motion and sodium voiding in subassemblies experiencing an overpower condition leading to significant fuel melting, fuel-pin failures and fuel ejection into unvoided or partially voided coolant channels. The degree of fuel-pin disruption is limited to cladding ruptures through which molten fuel can be ejected into the coolant channels. Thus, this model will be appropriate for treating the early post-pin-failure phase of higher ramp rate transient overpower (TOP) accidents and the entire post-failure phase of lower ramp rate TOPs if no complete pin disintegration or extensive cladding melting occurs. Of additional importance is the module's application to the early post-pin-failure fuel motion and sodium voiding in unvoided or partially voided subassemblies experiencing a Loss-of-Flow driven TOP (LOF'd'TOP). Once complete pin disintegration, high fuel vapor pressures, or molten cladding motion have to be considered, a switch to the LEVITATE module, which is designed to treat these phenomena, is made. This compatibility between PLUTO2 and LEVITATE is important for a consistent continuation of the analysis of the transient. This compatibility did not exist between the corresponding SAS3D modules SAS/FCI [14-1] and SLUMPY [14-2] which could never be used sequentially to treat the same fuel pin.

The PLUTO2 module can be used in all situations for which the SAS/FCI module of SAS3D was used. Moreover, the PLUTO2 fuel pins can fail into partially voided coolant channels, which was not possible with SAS/FCI. The models in PLUTO2 are considerably more mechanistic than those used in SAS/FCI. For example, fuel motion in SAS/FCI was largely treated with a lumped parameter approach, whereas PLUTO2 has one-dimensional models of the fuel motion inside the pins and in the coolant channels. Moreover, PLUTO2 treats phenomena that were not addressed in SAS/FCI such as continuous molten fuel flow regimes and the plateout of freezing fuel.

The PLUTO2 module is an outgrowth of the earlier PLUTO code [14-3, 14-4] which has been successfully used to simulate the early sodium voiding and fuel motion in TOP experiments [14-5, 14-6]. Therefore, the basic PLUTO concept has been retained in PLUTO2. However, as mentioned above, PLUTO2 is also designed to analyze the longer-term phenomena in mild TOPs and the early phases of LOF'd'TOPs. Moreover, PLUTO2 has an Eulerian hydrodynamic treatment which eliminates most of the cumbersome rezoning that was necessary in the Lagrangian PLUTO code. The Eulerian treatment has allowed the incorporation of axial cladding rupture propagation, fuel plateout, and variable cross section flow. Furthermore, PLUTO2 is considerably more economical to run than PLUTO; this is of importance for a module of a whole-core analysis code.

The PLUTO2 (and LEVITATE) coolant-channel treatment considers the entire subassembly cross section with all associated cladding and structure although the treatment is still one-dimensional. This is somewhat different from the concept that is used in the pre-fuel motion phase in SAS4A. The latter considers the coolant flow area, cladding and structure associated with only one pin.

Schematics of the PLUTO2 model when used for a mild TOP calculation and for a LOF'd'TOP condition are shown in Figure 14.1.1 and Figure 14.1.2, respectively. These schematics are for a single pin with a coolant channel that can belong to one or more pins. Although the PLUTO2 channel treatment is one-dimensional (i.e., there is a common treatment of all subchannels), not all of the fuel pins in a subassembly have to fail simultaneously. A number of the pins, as specified by input, can remain intact. A treatment of the delayed failure of these pins is not yet operational. In cases that assume that only some of the pins in a subassembly fail, the fuel and gas ejection from these failed pins will be added to all coolant subchannels since all subchannels in a subassembly are treated alike. This has prompted the use of the above-mentioned concept in PLUTO2 in which the basic channel cell has a cross section equal to the entire coolant flow area in a subassembly.

The PLUTO2 model addresses three major modeling areas pertinent to the post-pin-failure behavior. The first area is the in-pin fuel motion toward a cladding rupture, the second is the fuel and gas ejection through the cladding rupture, and the third is the multi-component, multi-phase hydrodynamics treatment in the coolant channel.

The basic assumption for the in-pin motion is the treatment of the molten fuel region inside the pin (see Figure 14.1.1 and Figure 14.1.2) as a pressurized cavity containing fuel and fission gas, which flow toward a cladding rupture. This general concept was already used in SAS/FCI, PLUTO, and EPIC [14-7, 14-8]. The key PLUTO2 features in the modeling of the in-pin flow are:

A1) Treatment of the two-phase, two-component flow as a compressible homogeneous flow with variable flow cross section and strong mass sinks (due to fuel ejection) and mass sources (due to the addition of melting fuel).

A2) Modeling of the radial and axial growth of the molten fuel cavity due to additional fuel melt-in determined by a heat-transfer calculation in the solid fuel annulus which is performed by subroutine PLHTR.

A3) The treatment of two types of fission-gas bubbles in the molten cavity. These are the large fission-gas bubbles on the grain boundaries which are assumed to act like free gas and exert pressure in the molten cavity and very small intragranular bubbles which do not affect the cavity pressure directly because they are assumed to be totally constrained by surface tension. However, these small bubbles can coalesce into large bubbles and thus contribute to the cavity pressurization in a delayed fashion. In the code, the rate of coalescence is controlled by an input time constant. Up to the time of pin failure, when PLUTO2 is initiated, the relative fractions of these two types of gas bubbles in the solid fuel and molten fuel cavity are calculated by DEFORM. DEFORM has the same two-bubble fission-gas treatment and uses the same input grain-boundary gas fraction and time constant for small bubble coalescence. Once PLUTO2 is active, additional fuel can melt into the cavity which brings fission gas with it. The total amount of fission gas in the melting fuel node is known from the steady-state DEFORM calculation. In the solid fuel, the DEFORM calculation tracks grain boundary gas (large bubbles) and intragranular gas (small bubbles) separately. The input fraction (FIFNGB) should roughly correspond to the fraction of gas on the grain boundaries in the solid fuel as determined from post-irradiation examinations of fuel pins.

A4) A simple fuel vapor pressure calculation which is based on the radially averaged temperature in a cavity node is performed. The fuel vapor pressure and the fission-gas pressure are assumed to be additive. A better fuel vapor pressure calculation would be based on the maximum fuel temperature in a node. However, computing the latter in a moving fluid is difficult. Also, it is not clear that any significant radial temperature gradients remain present after the onset of fuel motion. This problem will be addressed in future SAS4A work.

The fuel ejection from the pins is based on the assumption that the cavity pressure in the node behind the cladding rupture of the failed pins equilibrates instantaneously with the coolant channel pressure at the same elevation by ejecting appropriate amounts of fuel and gas. The latter are ejected with the same volume ratio as present in the cavity nodes. If the pressure in the coolant channel becomes temporarily higher than the pressure in the failed cavity nodes, the fuel and gas ejection is halted. Backflow of fuel, gas or sodium into the pins is not allowed. Two items concerning pin failure coherency and the axial pin-failure propagation are of importance:

B1) There is an option to fail only some of the pins in a subassembly. The other pins remain intact. This option is relevant for mild TOPs. Because of the common one-dimensional treatment of all subchannels, this requires

that the failed pins are reasonably well distributed over the subassembly cross section (see Figure 14.1.3). A problem with this option is that LEVITATE cannot be switched on after PLUTO2 because LEVITATE assumes that all pins in a subassembly are failed.

B2) The cladding failures can enlarge axially; this is also referred to as axial pin-failure propagation. There is a nonmechanistic pin-failure propagation option available which is keyed on input fuel melt fraction, cladding temperature, and required pressure difference between cavity and channel. The mechanistic option compares the calculated cladding hoop stress with an ultimate tensile strength function. Moreover, an input fuel melt fraction has also to be exceeded.

> The axial pin-failure propagation is of key importance for lower power channels experiencing a high overpower condition due to an LOF accident (LOF'd'TOP). In this case, the cladding can be relatively soft along a considerable length of the channel at the time of the initial failure. This may lead to rapid axial failure propagation. For milder overpower conditions, the axial failure propagation would be slower and mostly caused by the overheating of the cladding by molten fuel that has been ejected into the coolant channels.

In the coolant channels, PLUTO2 treats the flow of materials as a one-dimensional, compressible two-fluid flow with variable flow cross section. One component of this two-fluid treatment is the mobile fuel and the fission gas that is dissolved in the fuel. The other component is the mixture of liquid sodium, sodium vapor, fuel vapor, and free fission gas. The salient features of the channel modeling can be seen in Figure 14.1.1 and Figure 14.1.2 and are briefly discussed below.

C1) Coolant channel boiling is modeled by treating a static sodium film that is left behind by the expelled coolant slugs (see upper voided region in Figure 14.1.1 and Figure 14.1.2). The sodium film can be entrained by vapor streaming and by the action of fuel particles. Once the fuel flow regime becomes continuous (see below), the sodium film is instantaneously entrained as droplets in the gaseous phase. This sodium film is of importance because it provides a significant sodium vapor source and it cools the cladding surface for some time after sodium voiding has occurred.

C2) The liquid sodium slugs above and below the interaction region (which encompasses all the two-phase sodium, fuel, and fission gas) are treated as incompressible slugs with variable flow cross section. However, during the first few milliseconds after failure, an acoustic approximation is made to determine the slug interface velocities.

C3) Three different fuel flow regimes are treated in PLUTO2: particulate, partially or fully annular, and bubbly. The flow regime selection is mainly keyed to the liquid sodium fraction and to the channel fuel volume fraction.

The fuel motion in TOP accidents has been traditionally modeled as a particulate fuel suspension in a two-phase sodium/fission-gas mixture. However, the breakup of fuel into droplets or particles is likely only when the liquid sodium fraction is fairly high. From TREAT experiments, it can be concluded that continuous molten fuel flow regimes exist in voided regions. In PLUTO2, continuous fuel flow regimes are, therefore, considered in addition to the particulate flow regime. The treatment of a partially annular fuel flow regime is not a common approach but has been prompted by the notion that a relatively small amount of molten fuel in a voided channel will not cover the entire cladding and structure perimeter, but will rather behave like a single or multiple rivulet flow. For higher fuel fractions, a complete annular fuel flow regime is assumed and for an even higher fuel volume fraction, a bubbly fuel flow is modeled. Figure 14.1.3 illustrates the particulate and partially annular flow regime models for a seven-pin TREAT test bundle for the case in which only a certain fraction of the pins has failed. For the particulate or bubbly fuel flows, the fuel is simply uniformly distributed in all subchannels. For the partially annular flow, the fuel mass is assumed to be distributed between the pin cladding and structure in proportion to the cladding-to-structure surface area ratio. The cladding of all failed and unfailed pins is assumed to be covered by equal amounts of fuel film with equal film thickness. The fraction of the cladding perimeters covered by the fuel films is dependent on the fuel volume fraction and the input constant CIANIN. Where this fuel is exactly located on the pin perimeters is not relevant since there is no azimuthal cladding temperature distribution calculated. The fraction of the fuel covering the structure is of considerable importance in small bundles. Once frozen fuel crusts are generated (see below), they are also distributed in the same manner. Once the fuel fraction

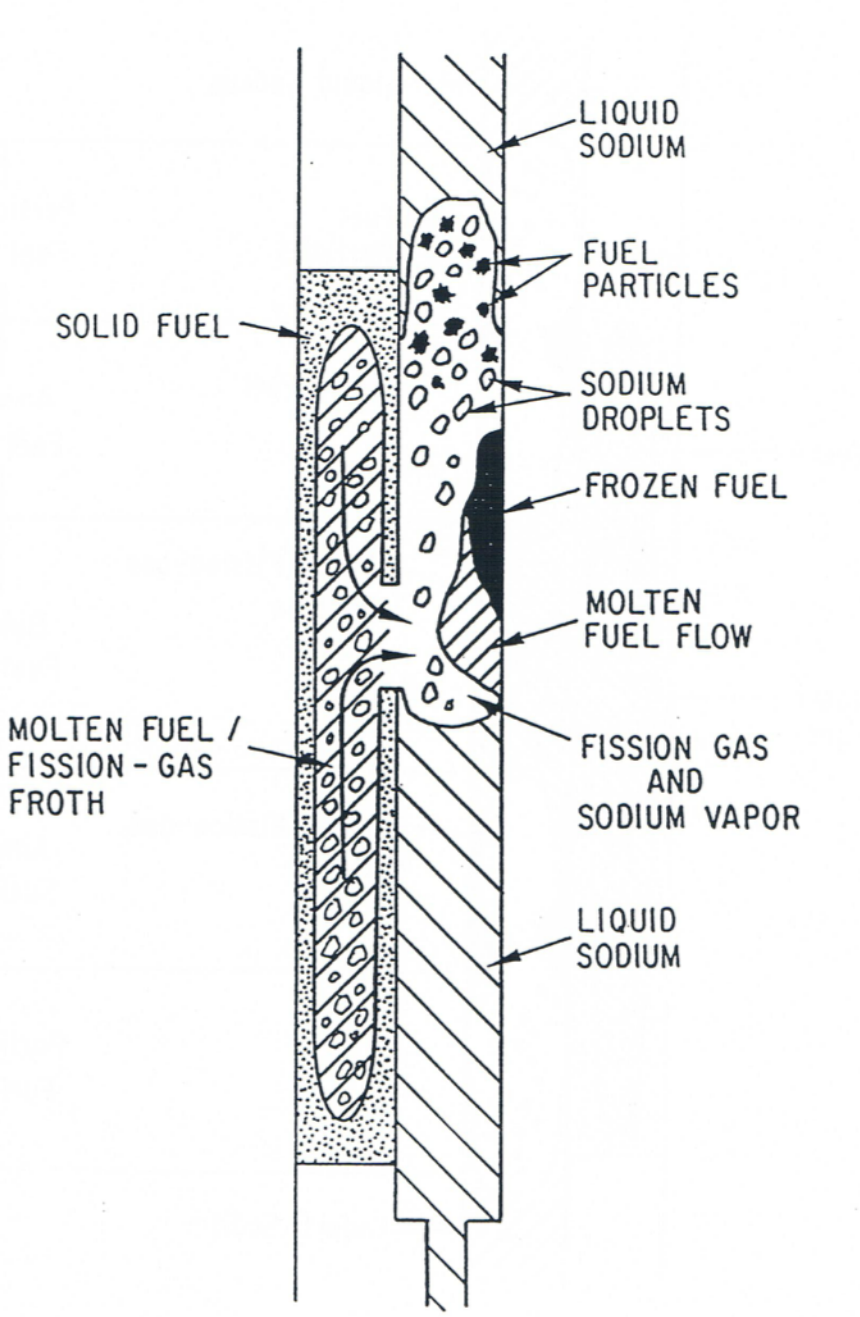


Figure 14.1.1: PLUTO2 Schematic for Transient Overpower Conditions

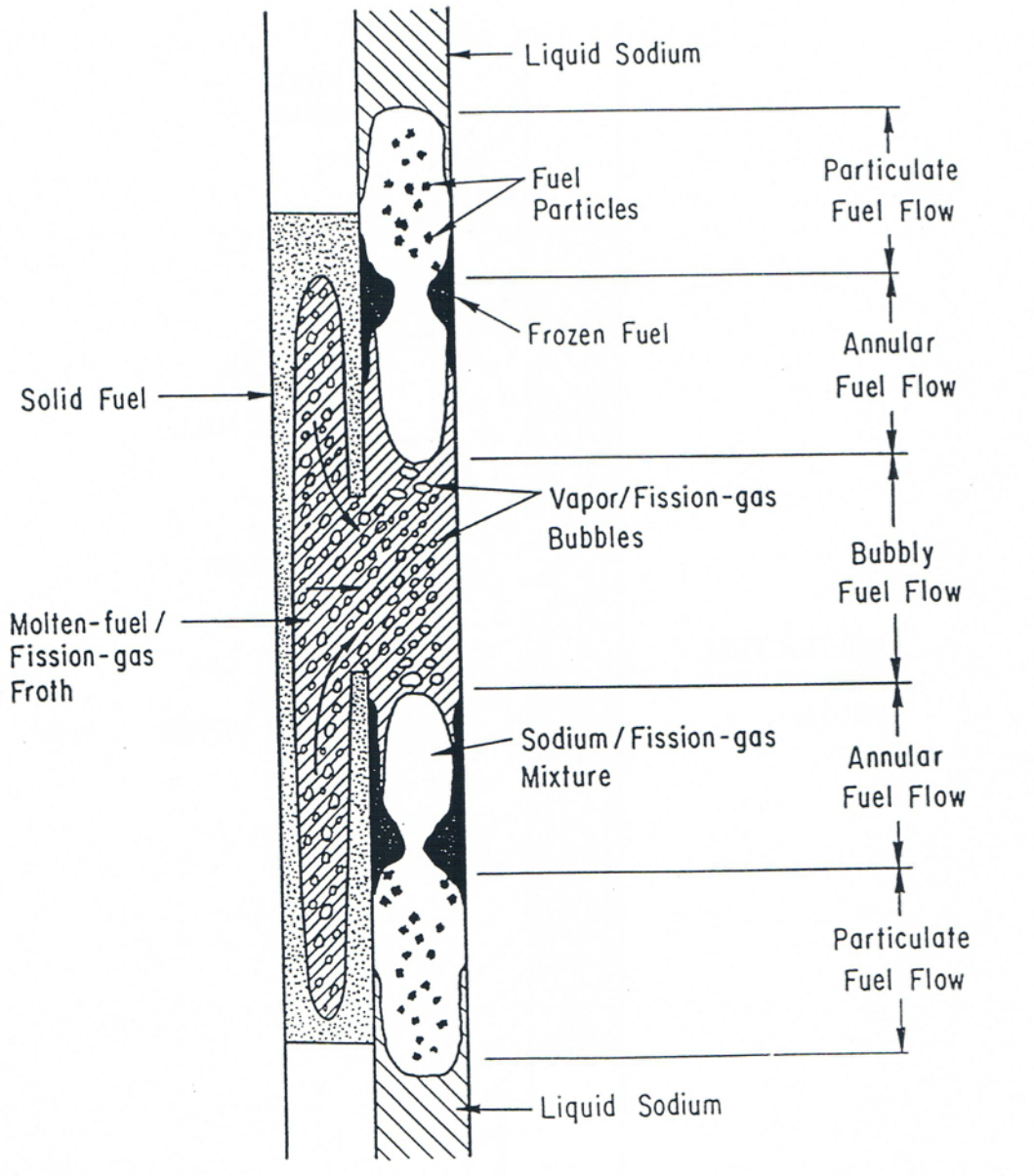


Figure 14.1.2: PLUTO2 Schematic for Loss of Flow Driven Transient Overpower Conditions

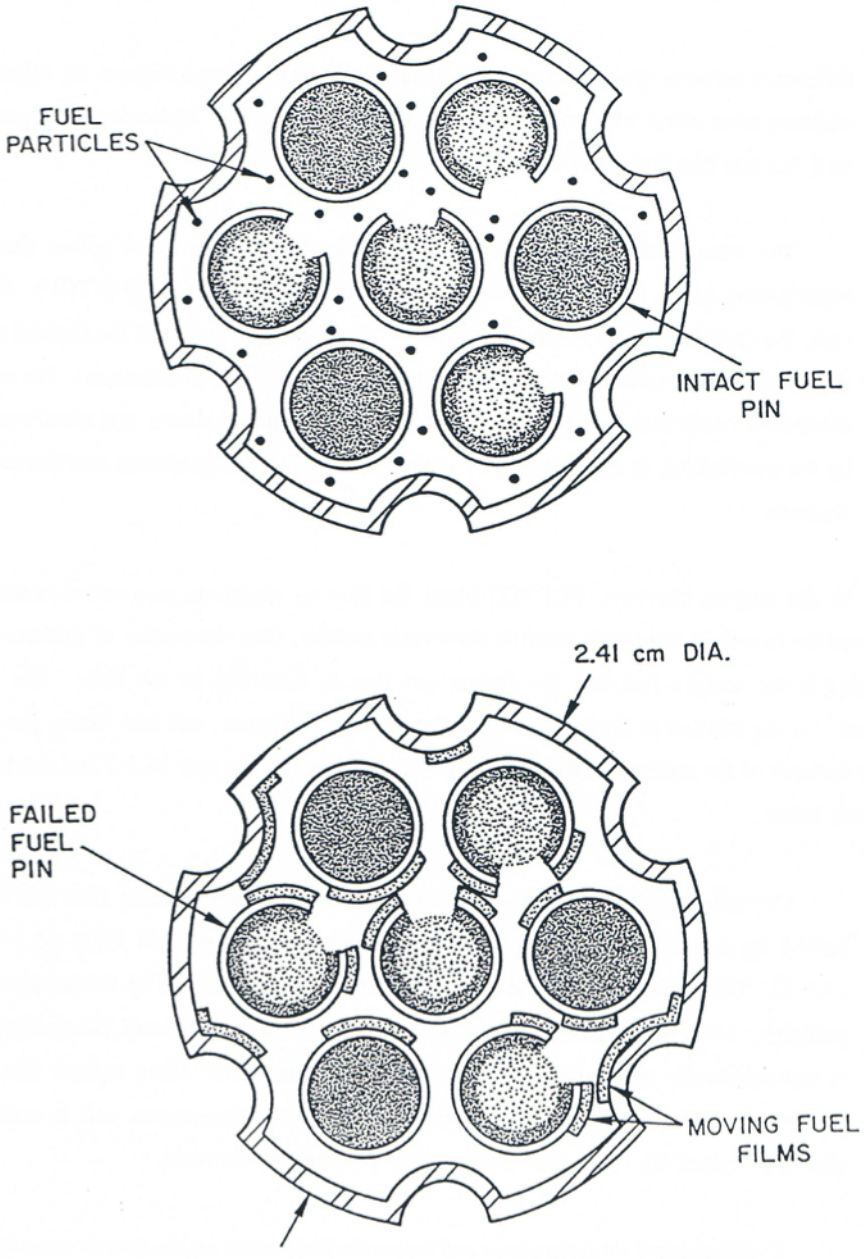


Figure 14.1.3: Illustration of Particulate Fuel Flow Regions (Upper Figure) and Partial Annular Fuel Flow Regime (Lower Figure) for a Seven-Pin Bundle in Which Only Four Pins Are Assumed to Have Failed

is high enough to lead to a fully annular flow, all cladding and structure in a given node is covered by the fuel films. The fuel flow regimes will be discussed in more detail in Section 14.4.2. Figure 14.4.1, which is shown later in Section 14.4, illustrates the fuel flow regimes in more detail for an equivalent cylindrical geometry.

Modeling these different fuel flow regimes explicitly has the advan-tage that one has all interaction areas for heat, mass, and momentum transfer readily available.

C4) Frozen fuel plateout is treated in PLUTO2 because there is over-whelming evidence from in-pile and out-of-pile experiments that fuel freezing and plateout are key phenomena. In PLUTO2, only fuel in the continuous flow regime can plate out on cladding and structure upon fuel freezing. The fuel particles are not allowed to stick to cladding and structure because the fuel particles are assumed to have a solid shell due to their interaction with liquid sodium.

> The modeling of the fuel plateout in PLUTO2 can either be of the bulk-freezing type or conduction-limited type. This is controlled by input parameter CIFUFZ, which also allows intermediate modes. Frozen fuel crusts can also become mobile after remelting. In addition, frozen fuel crusts are released into the moving fuel stream if the underlying clad or structure becomes significantly molten.

C5) The Fuel-Coolant Interaction (FCI) treatment depends on the fuel flow regime. In the particulate flow regime, the FCI treatment is largely based on the Cho-Wright approach [14-9] which considers the heat flow resistance in the fuel particles and ignores the resistance of liquid sodium. Moreover, the resistance due to vapor blanketing is treated in a parametric fashion in PLUTO2. In the code, separate FCI calculations are done for every numerical cell, whereas the Cho-Wright model is a lumped-parameter approach. Of importance for the FCI treatment in PLUTO2 is also the treatment of the slip between fuel and liquid sodium which mitigates the strength of the FCI's.

> In the annular fuel flow regimes in PLUTO2, the convective heat transfer between the hot fuel film and/or fuel crusts and the two-phase sodium/fission-gas mixture is considered. Since the fuel surface area for the annular flow is significantly smaller than that for the particulate flow, this type of FCI is much milder.
>
> In the bubbly fuel flow regime high heat-transfer rates between fuel and liquid sodium are possible, but the bubbly fuel flow regime is usually generated at an axial elevation where no liquid sodium is present. The penetration of liquid sodium into a bubbly fuel flow regime is also unlikely in the one-dimensional PLUTO2 because the cladding near a node with bubbly flow is usually too hot to allow reentry of liquid sodium.

The numerical grids on which the hydrodynamics equations are solved are shown in Figure 14.1.4. The stationary Eulerian grid in the molten pin cavity is aligned with the stationary Eulerian grid in the coolant channel. The cavity grid can expand continuously in the radial direction due to fuel melt-in and stepwise (by whole cells) in the axial direction. The grid on which the reactivity calculations are done covers the pin and blankets and the adjacent coolant channel cells from K = 1 to K = MZ. Outside the molten cavity region, this grid is also subdivided radially for the heat-transfer calculation in the solid fuel and cladding. The interaction region in the coolant channel can expand or contract continuously in the axial direction, which requires a partially Lagrangian treatment for the edge cells of the interaction region. The channel fuel region and the fission-gas region also expand or contract continuously in the axial direction. However, when they have moved into a new cell, the fuel or fission gas is assumed to be homogeneously distributed in that cell for calculating the pressure, heat transfer, momentum change, and reactivity. On the mesh grids above and below the interaction region, only the liquid sodium temperatures are calculated. The momentum change of the liquid sodium slugs is calculated in an integral fashion.

The current status of the PLUTO2 validation and its future validation needs are discussed in the SAS4A Validation and Verification Plan [14-10]. Here only an enumeration of the integral validation efforts already performed will be given. As mentioned earlier, the PLUTO code has been successfully used to simulate the early sodium voiding and fuel motion in two in-pile TOP experiments [14-5, 14-6]. PLUTO2 comparison calculations with

PLUTO showed good agreement for the early fuel motion and sodium voiding [14-11]. An intercode comparison with the EPIC code for LOF'd'TOP conditions showed that the two codes compare rather well when several of the advanced features in PLUTO2, such as fuel flow regimes and fuel plateout, are switched off [14-12]. PLUTO2 was also used in the EEC-WAC TOP comparative exercise [14-20]. Because its advanced features were active in this comparison exercise, the PLUTO2 results differed considerably from those calculated with simpler models.

A reasonably good post-test simulation of the major flow event in the H6 50 ¢/s TOP TREAT test [14-13] was achieved through input parameter adjustments [14-6]. However, some uncertainty with regard to the mode of the FCI observed in this test could not be resolved. A good post-test simulation of the L8 LOF'd'TOP TREAT Test [14-14, 14-15, 14-12] was also achieved after introduction of a model for frozen fuel crust release from molten cladding. This was prompted by a pre-test analysis which underpredicted the fuel dispersal [14-16]. Other pretest predictions with PLUTO2 were made for the AX1 $3/s TOP test using carbide fuel [14-17], the W2 10 ¢/s TOP test in the ETR [14-18], and the 37-pin bundle P4 pin-failure propagation test in ETR [14-19]. The prediction for the AX1 test was quite reasonable, whereas the prediction for W2 suffered from the assumption of too coherent pin failures. The PLUTO2 P4 pretest analysis predicted a complete sweepout of the fuel ejected from the three fuel canisters used in this test, whereas the experiment led to very little sweepout and a sizable frozen fuel blockage. Possible explanations of the observed behavior include sodium bypass effects in a large bundle or the fuel canister ejecting the fuel more violently than the fuel pins in TOP tests.

14.1.2 Overview of the Program Flow

Either PLUTO2 or LEVITATE is initiated once the subroutine FAILUR has predicted pin failure and a minimum fuel melt fraction equal to the input parameter FMELTM exists at the failure node. Subroutine FAILUR is usually called from the driver of the DEFORM pin-behavior module. However, in the case when cladding motion has already begun, it is directly called from the transient SAS4A driver routine TSTHRM. The decision as to which one of the two modules, PLUTO2 or LEVITATE, is to be initiated depends on the existence and size of a boiling region at the time of pin failure and whether the pins are predicted to fail into a voided or unvoided region of the coolant channel. If there is no boiling, as in an intermediate ramp rate TOP accident or if the pins fail into the liquid region of a partially voided channel, PLUTO2 will be initiated because of its capability of treating FCI's and sweepout of particulate fuel. For the case of fuel failing into a multibubble boiling region, PLUTO2 will only be initiated if the average void fraction in this region is less than 70%. If the average void fraction is larger or if cladding motion has begun, LEVITATE will be initiated.

The PLUTO2 initialization is described in more detail in Section 14.7.1. The subroutines involved in this procedure are PLINPT, PLSET, PLSETl, and PLSAIN. In these routines all permanent PLUTO2 variables are either set to input values or values calculated by the single-phase hydraulics, the pin heat-transfer routines, the pin behavior module DEFORM, or the boiling module. It should also be mentioned here that the flag ICALC, which designates which major module is currently active, will be set to 3 if PLUTO2 has been initialized and to 2 if LEVITATE has been initialized.

Once the PLUTO2 initialization routines have been executed and the flag ICALC has been set to 3, the transient SAS4A driver TSTHRM (see flow diagram in Chapter 2) will call PLUDRV (*PLU*TO2 *DR*IVER SUBROUTINE). PLUDRV will retain control and advance the solution using PLUTO2 time steps until the end of the primary-loop time step is reached. If another SAS4A channel is using PLUTO2 at this time, its solution will also be advanced until the end of the current primary-loop time step.

The flowcharts in Figure 14.1.5 and Figure 14.1.6 show the logic of the PLUTO2 driver. They are complete except for an option for switching off all subroutines performing the material velocity calculations. This option is discussed later in this section. If the flag MODEPL, which controls this option, is zero, the flow diagrams in Figure 14.1.5 and Figure 14.1.6 are appropriate.

If the flag ILEPLI has been set to 1 in the PLUTO2 initialization (see Figure 14.7.2 in Section 14.7), LEVITATE will have to be initiated via PLUTO2. In this case, the PLUTO2 driver calls LEPLIN which is the interface

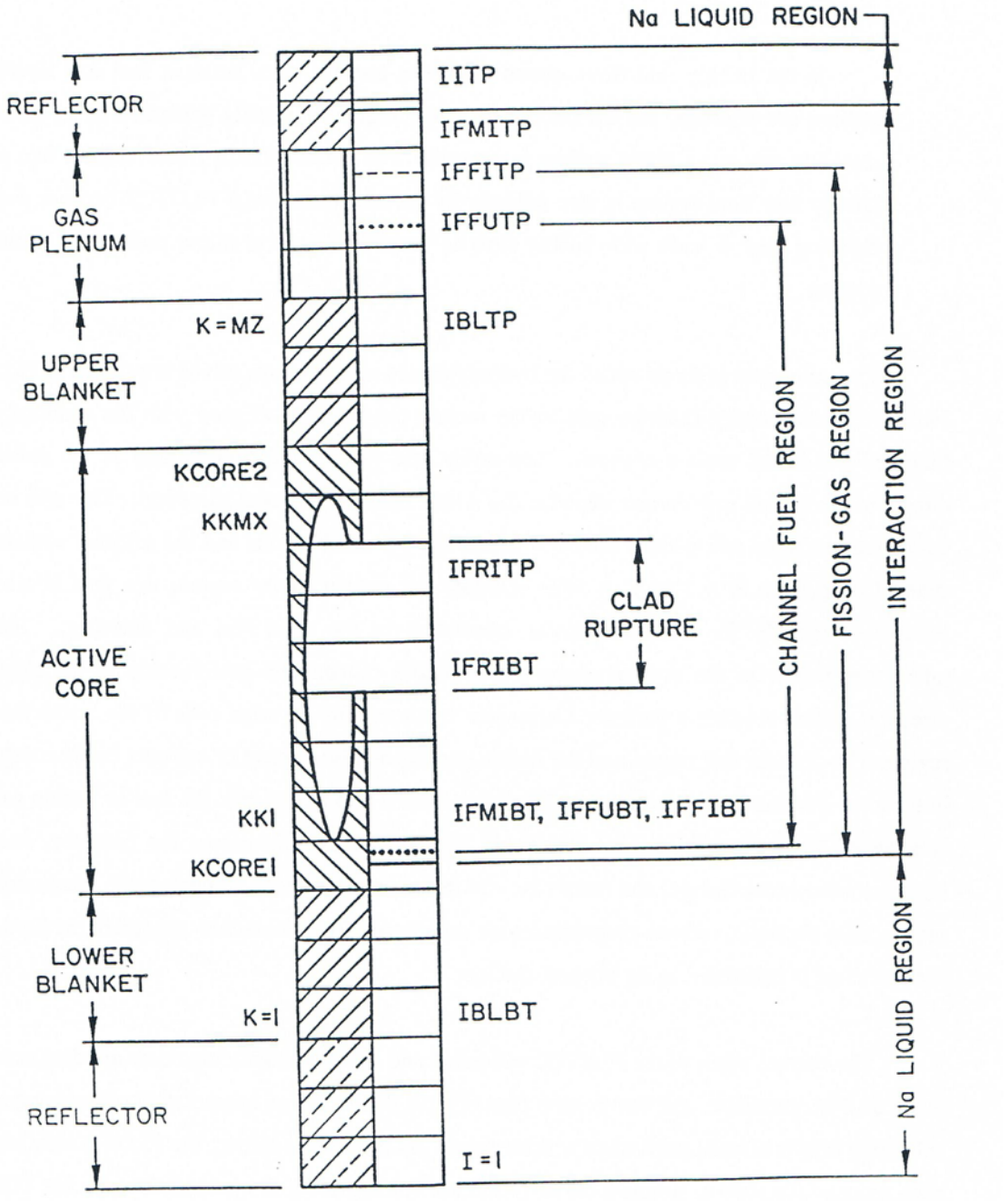


Figure 14.1.4: Axial Mesh Cells Used in the Pin and the Coolant Channel for the Numerical Solution of Conservation Equations. Also Shown Are the Different Component Regions in the Channel

routine between PLUTO2 and LEVITATE. The flag ICALC is then set to 2 and control is returned to TSTHRM. The latter will call the LEVITATE driver at the beginning of the next coolant time step upon encountering ICALC = 2 for the channel under consideration (see flow diagram of TSTHRM in Chapter 2).

If ILEPLI is not equal to 1, subroutine PLSET2 will be called (see Figure 14.1.5). In this subroutine, all temporary arrays (i.e., arrays which can be overwritten once control is returned to TSTHRM) are initiated. Moreover, temporary integers are set.

Next, the PLUTO2 time is advanced by adding the PLUTO2 time-step size to the time at the beginning of the PLUTO2 cycle. The time-step increment for the very first PLUTO2 cycle is an input value which is also the minimum PLUTO2 time-step size. Later, in the logic flow, the time-step size is calculated.

Next, the inlet and outlet pressures at the end of the PLUTO2 time step are determined from the PRIMAR calculated inlet and outlet pressures and rates of inlet and outlet pressure changes.

Next, the power level at the end of the PLUTO2 time step is calculated from an exponential fit of the power-time history that takes the power level at the beginning of the previous and the current main time steps and the precalculated power level at the end of the current main time step into account. By using the calculated power level and the axial input power distribution, the specific power for each axial pin node is calculated. The same specific power is set for the corresponding channel nodes.

In subroutine PSHAPE, which is not a PLUTO2 subroutine, the total power for all axial fuel-pin nodes is calculated assuming that the pin is still intact. This is needed for the heat-transfer calculation in the solid fuel annulus in the pin.

Next, cladding and structure temperatures in the nodes adjacent to the interaction region are initialized in order to make sure that they will be available if the interaction zone expands into one of these nodes.

In subroutine PLIF (*PL*UTO2 *I*NTER*FA*CES), the axial displacements of the sodium slug interfaces, the interfaces of the fission-gas region and the interfaces of the regions containing fuel are reset. The actual calculations of the velocities needed for this resetting are performed later in subroutine PLMOCO. The initial sodium velocities come from the single-phase hydraulics or the boiling model and are set in PLSAIN. Fission gas and mobile sodium velocities are always the same in PLUTO2. Subroutine PLIF also calculates the axial pin-failure propagation and resets the pointer array IDISR(I) which indicates which pin nodes have failed.

In subroutine PLREZO (*PL*UTO2 *REZ*ONE), mesh cells are added to an expanding interaction region or deleted from a shrinking interaction region. Moreover, it cuts off short columns of liquid sodium slugs which are attempting to reenter into a cell of the interaction zone containing frozen fuel or ruptured cladding. The liquid sodium is added homogeneously to such cells. If fuel pins fail into the lower sodium slug, PLREZO will enlarge the interaction region downwards.

In subroutine PLFREZ (*PL*UTO2 *FRE*EZING ROUTINE), the amount of fuel plating out per time step and per node is calculated. Moreover, PLFREZ calculates the amount of crust released because of remelting or because the underlying cladding has become molten. The released fuel crusts, which have an axial velocity of zero, are mixed with the moving fuel and an updated velocity is calculated by momentum averaging.

In PLMACO (*PL*UTO2 *MA*SS *CO*NSERVATION), the mass conservation equations for the moving components in the channels are solved. This includes a combined mass conservation for solid or liquid fuel and fuel vapor, and mass conservations for the sodium, free fission gas and dissolved fission gas. There is a special treatment for the top and bottom cells of the channel. Fuel, sodium and fission-gas will be taken out and stored in a reservoir if the interaction region has extended into the lowermost or uppermost channel cell.

In subroutine PLVOFR (*PL*UTO2 *VO*LUME *FR*ACTIONS), the entrainment of the static sodium film by the flow of the two-phase sodium/fission-gas mixture and of the fuel particles as well as the de-entrainment of liquid droplets onto the film, is calculated. PLVOFR also sets the volume fractions of the various components based on the results of the plateout and crust release calculation in PLFREZ and the results of the mass conservation equations. The final section of subroutine PLVOFR is devoted to the selection of the fuel flow regime.

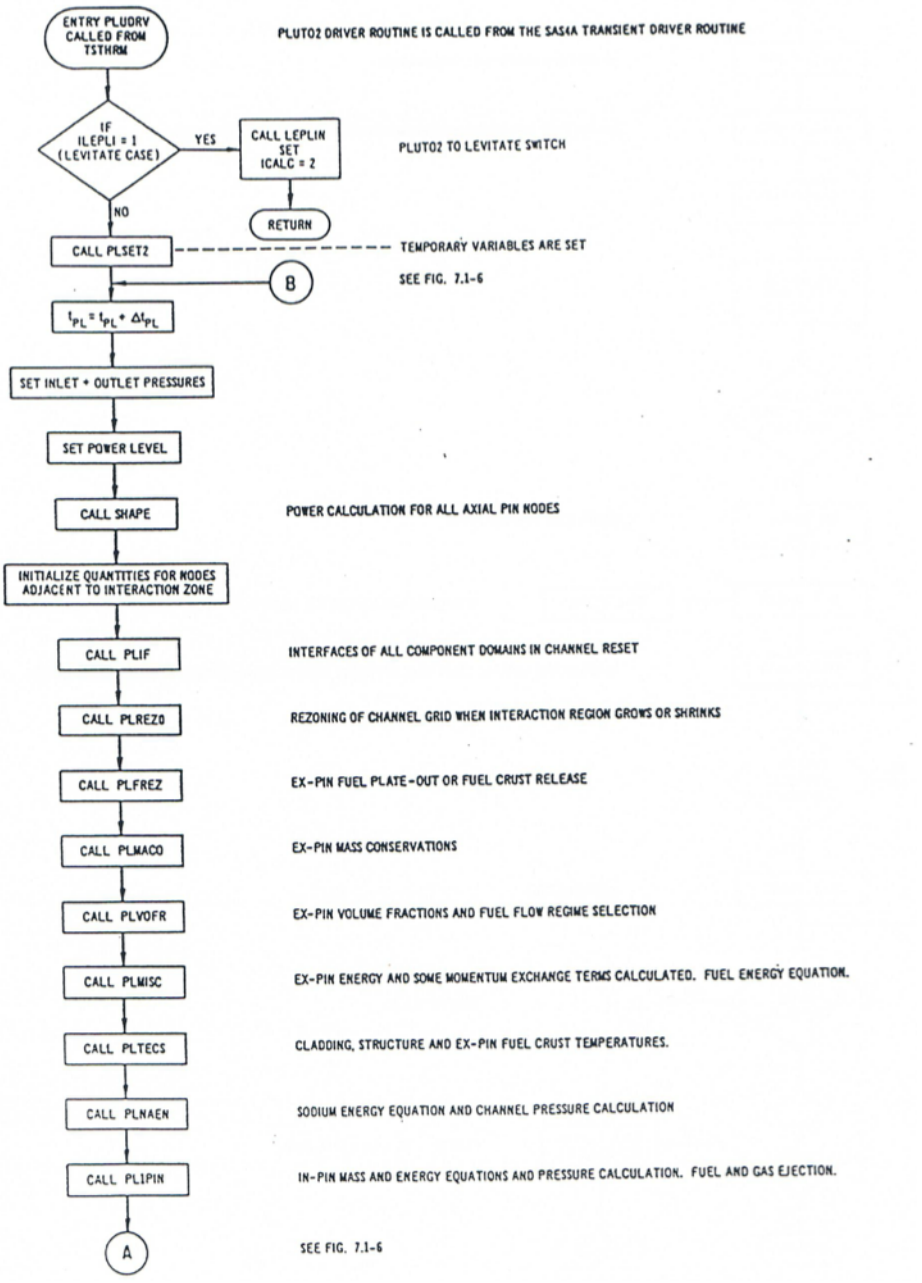


Figure 14.1.5: First Part of the Flow Chart of the PLUTO2 Driver Subroutine, PLUDRV, Showing the Functions of the Major Subroutines

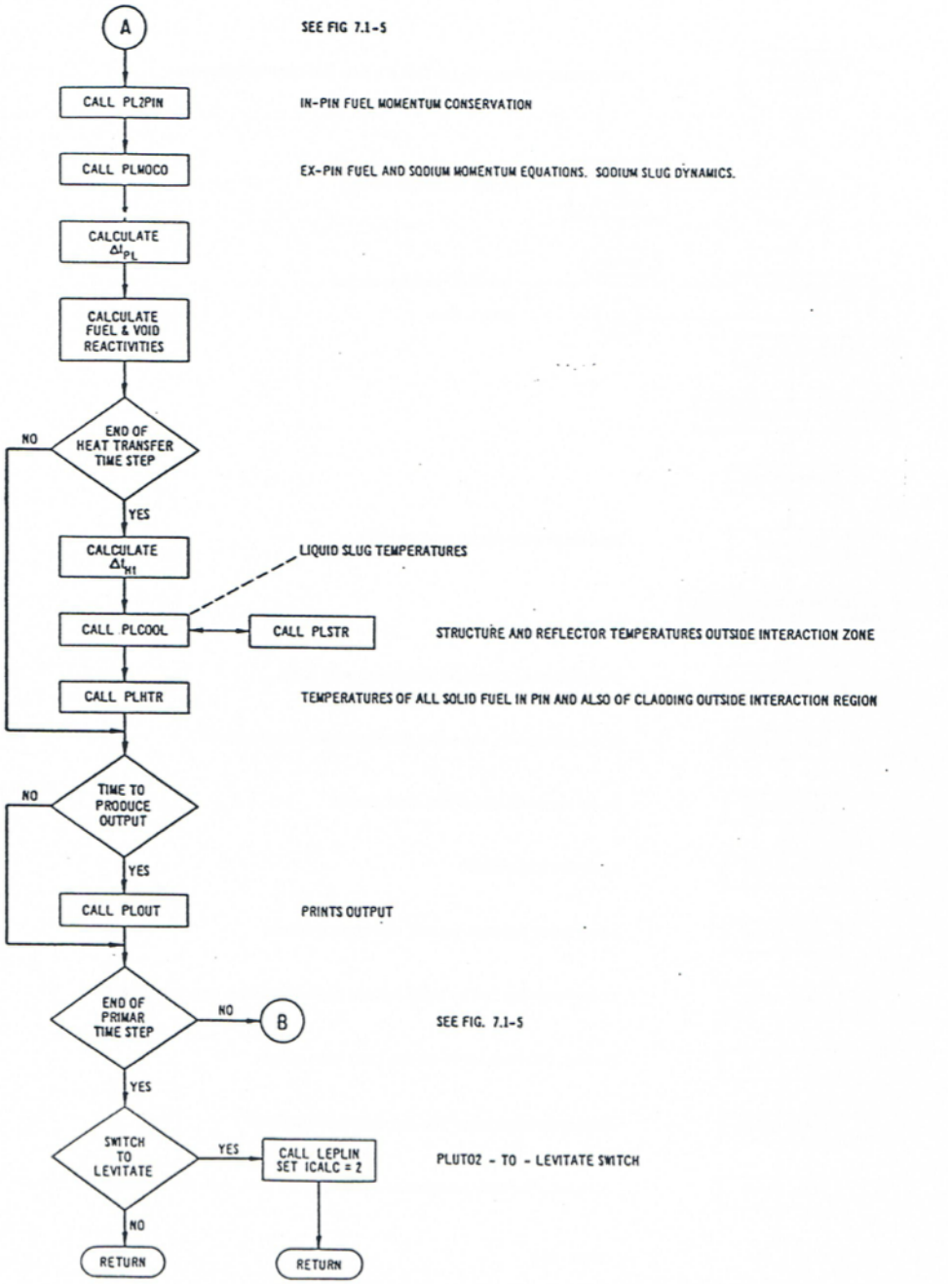


Figure 14.1.6: Second Part of the Flow Chart of the PLUTO2 Driver Subroutine, PLUDRV, Showing the Functions of the Major Subroutines

In subroutine PLMISC (*PL*UTO2 *MISC*ELLANEOUS), several important items are calculated. First, the molten- and frozen-fuel configurations in the flow channel are determined (see Figure 14.4.1). Second, most energy and some momentum exchange terms between the various flow components, cladding and structure are calculated. Because many of these interaction terms depend on the fuel flow regimes, three different exchange coefficients are needed for several of the components. Third, the mobile fuel energy equation is solved.

In subroutine PLTECS (*PL*UTO2 *TE*MPERATURE CALCULATION OF *C*LADDING AND *S*TRUCTURE}, the cladding, reflector, and structure temperatures within the interaction zone are calculated. This calculation is preceded by the determination of the energy exchange coefficients for the frozen fuel crust and the solution of the frozen crust energy equation. Since the energy exchange between fuel crust and moving fuel was not considered in subroutine PLMISC, an updating of the moving fuel energies is also performed here.

In subroutine PLNAEN (*PL*UTO2 *NA EN*ERGY EQUATION), the two-phase and single-phase enthalpy equations for the mixture of sodium and fission gas are solved. From the resulting temperatures and the previously calculated volume fractions (see PLVOFR), the liquid-phase sodium pressure or the sodium saturation or superheated sodium vapor pressure, as well as the free fission gas pressure, are calculated. In addition, the total end-of-time-step pressure, which includes the fuel vapor pressure, is calculated.

In subroutine PLlPIN (*PL*UTO2 NO.*1 PIN* EQUATIONS}, the mass and energy equations for the in-pin fuel motion are solved. At first, the fuel and free and dissolved fission-gas mass sources due to fuel melt-in are calculated and the in-pin cavity enlargement is determined. Moreover, the free fission-gas mass sources due to dissolved gas coalescence and the corresponding sinks for the dissolved gas are evaluated. Following these, the fuel mass and energy conservation equations and the free fission-gas mass conservation equation are solved. Then, a preliminary end-of-time-step pressure is calculated for all cavity cells. This pressure calculation takes the prior-calculated end-of-time-step densities and temperatures into account. Only the pressures in the ejecting nodes will be further updated; in the other cells, the "preliminary end-of-time-step pressure" is the pressure that will actually be used in the momentum equation. The last major item in PLlPIN is the calculation of the mass of fuel and fission gas ejected from all cavity cells that have failed cladding and a pressure higher than that in the corresponding channel cell. The masses of fuel and free and dissolved fission gases in these cavity cells are then reduced and the pressure is updated. The fuel and free and dissolved fission-gas masses in the channel cells are correspondingly increased and the end-of-time-step pressures in the channel nodes receiving fuel and/or fission gas are updated. The fuel energy in the channel nodes receiving fuel is also updated due to the addition of fuel with a higher temperature.

In subroutine PL2PIN (*PL*UTO2 NO. *2 PIN* EQUATIONS), the momentum equation for the homogeneous fuel/fission gas mixture in the molten pin cavity is solved. Next, the mass conservation equation for the dissolved fission gas is solved. This could have been solved earlier in PLlPIN but that routine had become too crowded. The calculation of the time-step size for the in-pin calculation during the next time step is also performed in PL2PIN.

In subroutine PLMOCO (*PL*UTO2 *MO*MENTUM *CO*NSERVATION), many quantities which were previously needed only at the cell centers have to be defined at the cell edges. Moreover, most of the momentum exchange terms and, in particular, the drag between the fuel and the two-phase sodium/fission-gas mixture are evaluated for the particulate and bubbly fuel flow regimes. The main section of this routine deals with the simultaneous solution of the two momentum equations at all cell edges in the interaction region. Moreover, momentum equations will also be solved for fuel particles at either end of the fuel region if the end nodes of the fuel region are in a particulate flow regime. Another section of PLMOCO deals with the calculation of the interaction zone interface velocities. This includes an acoustic approach in the liquid slugs during the first few milliseconds after pin failure and later an incompressible variable-cross-section treatment of the upper and lower coolant slugs. Also calculated in PLMOCO are instantaneous coolant slug flow rates for the entire calculational channel, as well as the integrated channel flow rates over a PRIMAR time step. These quantities are needed by the PRIMAR4 module for recalculating the inlet and outlet pressures.

The next task in the PLUTO2 driver routine is the time-step size determination in the coolant channel. This is compared with the time-step size calculated for the in-pin motion and the smaller of the two will be the PLUTO2

time-step size for the next calculational step.

The next task of PLUDRV is to calculate the transient mass distributions for the fuel and voiding reactivities. This calculation will be described in the next section on the interaction with other SAS4A modules.

If the time at the end of a PLUTO2 time step coincides with the end of the heat-transfer time step, several heat-transfer routines are called. This "coincidence" is forced to occur whenever the calculated PLUTO2 time step would overshoot the end of the heat-transfer time step. In this case, the PLUTO2 time step is set to coincide with the end of the heat-transfer time step. This is actually done right after the time-step size calculation described above.

The heat-transfer time step is determined next. Its maximum value is based on a characteristic heat-transfer time of the cladding and can be further limited by small primary-loop time steps which will be small in high power situations because the main (or point kinetics) time step becomes small. Since the heat-transfer time step is also used for the liquid slug temperature calculations, it can be further limited by a Courant condition based on the slug velocities.

In subroutine PLCOOL (*PL*UTO2 *COO*LANT SLUG TEMPERATURE), the temperature in all numerical nodes in the liquid sodium slugs is calculated. This involves the calculation of heat-transfer exchange terms between liquid sodium and cladding, plenum cladding, reflectors, and structure, and the solution of an energy equation. Moreover, PLCOOL checks whether a node in the lower sodium slug has started to boil.

In subroutine PLSTR (*PL*UTO2 *STR*UCTURE TEMPERATURE CALCULATION), the temperatures of the structure, the plenum cladding, and the reflectors outside of the interaction region are calculated.

In subroutine PLHTR (*PL*UTO2 *H*EAT *TR*ANSFER), calculations are made of the temperature fields in the solid fuel annulus surrounding the molten fuel cavity, in the unmelted fuel and blanket cells, and in the fuel and blanket cladding outside the interaction region. The heat-transfer boundary condition at the interface between the molten cavity and the solid fuel annulus is treated by applying a time-integrated heat flow rate term whose contributions were calculated and summed up in subroutine PLIPIN. The boundary condition between the inner cladding and the outer fuel surface is also based on such an integrated heat flow rate term. The latter is calculated and summed up in subroutine PLTECS, in which the cladding temperature field in the interaction region is calculated using the PLUTO2 time step.

If it is time to produce output, subroutine PLOUT (*PL*UTO2 *OUT*PUT) will be called. If the end of the PRIMAR time step has not yet been reached, the logic flow will return to point B (see Figure 14.1.5).

If the end of the PLUTO2 time step coincides with the end of the primary loop time step, a check is made whether the conditions require a switch to the LEVITATE module. This will be necessary if extensive cladding melting has occurred or if complete fuel-pin disruption is imminent or if the fuel vapor pressures have become quite high. In this case, the LEVITATE-PLUTO2 interface routine LEPLIN is called and the integer flag ICALC is set to 2 which will assure the calling of the LEVITATE driver routine LEVDRV at the beginning of the next PRIMAR step. Whether a switch to LEVITATE is made or not, control will now be returned to the transient driver TSTHRM.

An option, which is not shown in the flow charts in Figure 14.1.5 and 14.1-6, allows the user to set all material velocities to small values and to shut off all the subroutines or subroutine sections that are calculating the motion of materials in PLUTO2. This option, which allows an economical but very simplistic continuation of a PLUTO2 calculation for several tens of seconds, is useful for the later treatment of a lead channel failing long before other channels in a low-ramp-rate TOP calculation. This situation is most likely to occur when the negative reactivity introduced due to fuel sweepout from the lead channel is not enough to insure permanent subcriticality. This option may be reasonable for subassemblies that are completely blocked by frozen fuel as long as they do not lose much heat to neighboring subassemblies (this subassembly-to-subassembly heat loss should eventually be modeled in SAS4A). Moreover, if the fuel in the disrupted assembly heats up too much, it could also become mobile again; this is not currently treated. The above-mentioned PLUTO2 option will be activated if the input time TIPLMX is exceeded by the PLUTO2 time.

14.1.3 Interaction with other SAS4A Modules

During the PLUTO2 initialization, data from several other modules are transferred to PLUTO2 (see Figure 14.1.7). When PLUTO2 is used for a TOP calculation, it requires data from the single-phase hydraulics, the DEFORM pin behavior module, and the pin heat-transfer module for non-boiling conditions (whose driver routine is TSHTRN). When PLUTO2 is used in a SAS4A channel experiencing a LOF'd'TOP condition, data may also be needed from the boiling module and the heat-transfer module for boiling conditions (whose driver routine is TSHTRV). The PLUTO2 initialization has already been briefly discussed at the beginning of Section 14.1.2 and will be described in more detail in Section 14.7.1.

Once PLUTO2 is active, there will only be an interaction with the point kinetics module and also with the primary loop module if the PRIMAR-4 module has been selected (see Figure 14.1.7). Moreover, there is an interaction with the PLHTR pin heat-transfer calculation which is virtually a PLUTO2 routine but uses a different time step than PLUTO2 (see Section 14.1.2). There is no interaction with other modules because PLUTO2 calculates the motion of the liquid sodium slugs (in PLMOCO), has the capability to treat sodium boiling in regions which are not yet occupied by fuel, and also has a simplified calculation of the axial cladding failure propagation (in PLIF).

When PLUTO2 is active, it uses the user input axial fission power distribution with the magnitude calculated by the point kinetics module. PLUTO2 provides the point kinetics module with the sodium and fuel axial mass distributions for all channels in which PLUTO2 is active. Moreover, the Doppler reactivity calculation is based on PLUTO2 calculated average fuel temperatures. The details of these calculations are given in Section 14.6.1.

If the PRIMAR-1 option is chosen PLUTO2 will use a constant outlet coolant plenum pressure which is input and an inlet coolant plenum pressure which is calculated by PRIMAR-1 at the beginning of the transient and later modified by an input table PLUTO2 will not feed back and information to the primary-loop module if the PRIMAR-1 option has been chosen.

If the PRIMAR-4 option has been chosen, PLUTO2 uses the time-dependent inlet and outlet plenum pressures and temperatures which are calculated by PRIMAR-4. PLUTO2 feeds back to the PRIMAR-4 module the sodium masses ejected into or received from the inlet and outlet plena during a PRIMAR time step. PLUTO2 also provides the PRIMAR-4 module with the energy of the sodium ejected into the plena and with the liquid sodium flowrates at the end of the PRIMAR time step. This is described in more detail in Section 14.6.2.

The PLUTO2 calculation will be taken over by the LEVITATE module if extensive cladding melting has occurred or if complete pin disruption is imminent or if the fuel vapor pressure becomes quite high. This transition, which can be controlled by input, is necessary because PLUTO2 is not designed to treat these situations. However, the transition to LEVITATE will not occur if only some of the pins have failed in PLUTO2. This PLUTO2 option, which is useful for mild TOP conditions, causes problems for LEVITATE because the latter assumes that all pins have failed when it is initiated.

14.2 In-Pin Fuel Motion

14.2.1 Overview and Assumptions

When LMFBR fuel pins melt in an overpower accident, the interiors of the pins melt first and form cavities containing molten fuel and fission gas. Some of this fission gas is dissolved in the molten fuel and the remainder is free fission gas which resides in bubbles too large to be constrained by surface tension. The decrease in the void volume, caused by the density decrease of the melting fuel and also of the molten fuel, which is further heated, compresses the free fission gas and causes a hydrostatic cavity pressurization which loads the cladding. This cavity pressure increases further when more free gas becomes available due to additional fuel melting and release of dissolved gas, and when fuel vaporization occurs. This leads eventually to cladding failure. The subsequent fuel and fission-gas ejection from the fuel pin and the local depressurization in the cavity caused by this ejection leads to fuel motion inside the pin toward the failure location. If the pin failure location for driver

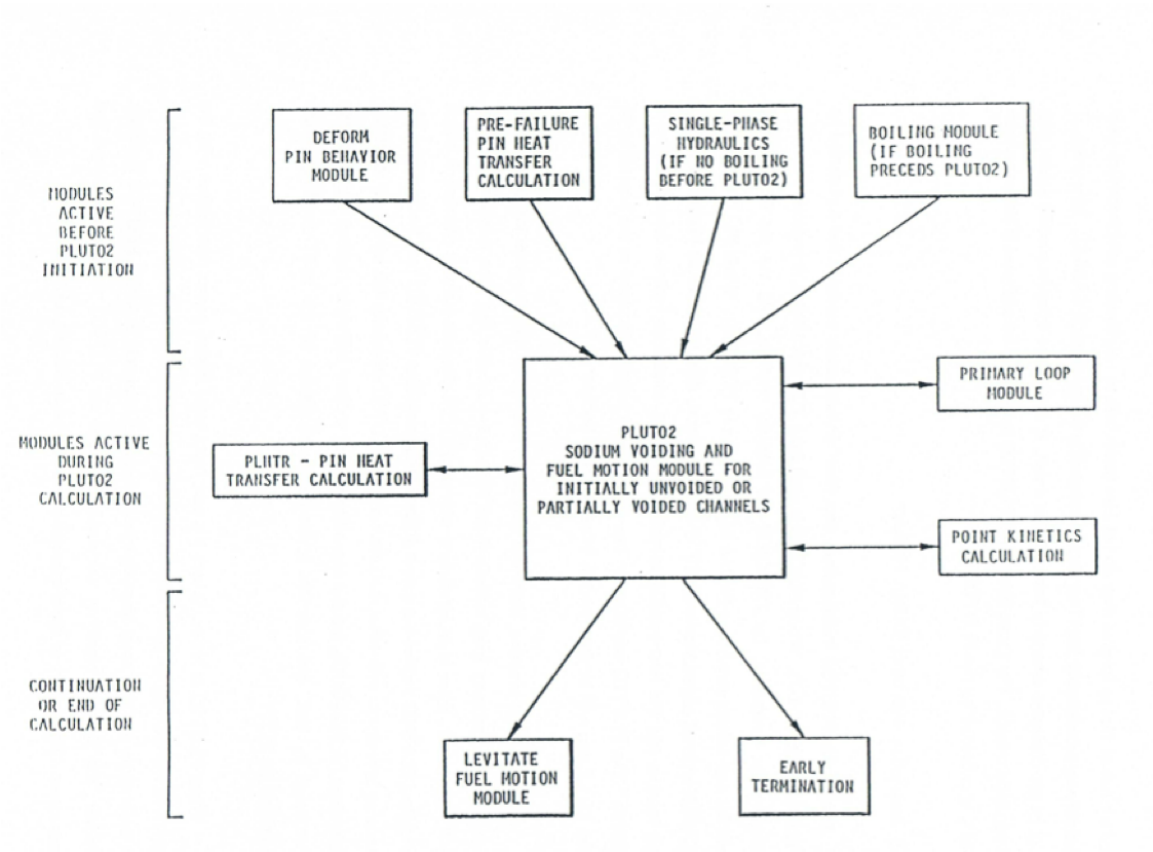


Figure 14.1.7: Data Transfer Between the PLUTO2 Module and the Other SAS4A Module

fuel is above the core midplane, this in-pin fuel motion reduces reactivity; if it is near the midplane, it can cause a reactivity addition.

The concept of a pressurized molten pin cavity, which was originally developed for TOP accidents, is also reasonable for LOF accidents which lead to a power rise and fuel melting before the cladding melts. Therefore, this cavity concept is used not only in PLUTO2 but also in LEVITATE. However, in an LOF accident, leading only to a few times nominal power or less, cladding motion and fuel swelling will precede fuel melting. In this case, the fuel pin is likely to disrupt completely into a mixture of molten fuel, gas, and solid fuel chunks once fuel melting has begun. This is only modeled in LEVITATE.

The in-pin fuel motion in the molten pin cavity in both PLUTO2 and LEVITATE is treated as a one-dimensional, compressible flow with a time dependent and spatially variable flow cross section. The fuel/fission gas flow is considered to be homogeneous and the fission gas is assumed to be in thermal equilibrium with the fuel.

Special attention had to be given to the potentially large mass sinks due to fuel and fission-gas ejection from the molten pin cavity. Mass sources due to fuel melt-in which cause flow cross section changes are also considered. The fuel which is melting into the cavity brings with it dissolved (intragranular) and free (intergranular) gas. These two types of gas are treated separately in the PLUTO2 and LEVITATE in-pin motion models. The free gas is considered not to be constrained by surface tension and to act as an ideal gas that is at the local fuel temperature. The dissolved gas, which is in the form of small bubbles, is assumed to be strongly constrained by surface tension, and its contribution to the local pressure depends on the input variable PRSFTN. If PRSFTN $\leq 10^7$ then the model assumes that the volume of the small bubbles containing the dissolved gas is negligible and thus the dissolved gas does not immediately affect the local pressure. If PRSFTN $>10^7$, the pressure calculation takes into account the volume of the dissolved gas bubbles, as described in Section 14.2.6. The dissolved gas is released (coalesces) according to a decay type law using an input decay constant. The amount released during each time step is added to the free gas. The fission-gas pressure calculation takes the compressibility of the liquid fuel into account and reduces to a liquid-fuel single-phase pressure calculation for no fission gas and zero void volume. The total pressure includes the fission-gas pressure and fuel vapor pressure. The fuel vapor pressure is based on the radially averaged fuel temperature.

Incoherency of the pin failures is mainly an issue in lower ramp rate TOP accidents. In these accidents, pin failure incoherency should be helpful for the post-accident coolability. If not all pins fail in the lead assemblies during a slow TOP, some subchannels may remain open and provide coolant paths. However, if near-midplane failures are assumed in slow TOPs in higher void worth cores, there may be a potential for exacerbating the accident due to pin failure incoherency. This is because the first pin failures will probably lead to some rapid fuel sweepout but may also cause significant voiding. Pins failing later will inject fuel into partially voided regions and this fuel may not experience rapid sweepout.

There are two major reasons for the incoherency in pin failures. One is the stochastic nature of pin failures and the other is due to differences in the power, flow, or coolant temperatures experienced by different pins. For example, the outer two fuel-pin rows in a subassembly see colder coolant temperatures due to the proximity of the hexcan wall. Many subassemblies in an LMFBR also see considerable radial power skews. In PLUTO2 an attempt has been made to treat the stochastic pin failures by allowing different pin failure groups. However, currently only one pin group can fail, and the other two groups of pins have to remain unfailed; (see input variables NRPIl, NRPI2, and NRPI3). The treatment of the delayed fuel expulsion and in-pin fuel motion for the two additional pin groups is not yet operational.

14.2.2 Initial Conditions for the In-pin Calculation from DEFORM and the Pre-Failure Pin Heat-transfer Calculation

Before pin failure, the molten fuel/fission-gas cavity is treated by the DEFORM pin behavior module. In the DEFORM treatment, the initially present central fuel hole is considered and the closing of this central hole due to the expansion of the molten fuel is calculated. Although DEFORM does not calculate axial fuel redistribution inside the molten cavity, an axially uniform pressure is assumed for the entire molten cavity and found by considering the total gas and volume available. (An option is also available in DEFORM to use the nodal pressures for loading the cladding. This option is relevant only for very high power situations in which the molten cavity grows to a large size within a few milliseconds.) When pin failure is predicted by DEFORM or because an input melt-fraction or failure time have been reached, PLUTO2 is initialized from DEFORM conditions. These include the uniform cavity pressure (or the axial pressure distribution if the special DEFORM option was used), the dissolved and free fission-gas densities, and the molten cavity geometry. However, PLUTO2 will have exactly the same cavity geometry as DEFORM only if the PLUTO2 input option *FNMELT* = 0.0 is chosen. This value means that the radial cavity boundary is defined by the location of the fuel solidus temperature (an assumption in DEFORM). However, it is recommended for PLUTO2 to use a higher melt fraction *(FNMELT* > 0.5) for the definition of the location of the cavity boundary. This is because the analysis of TREAT test E8 [14-5, 14-6] showed that fuel, which had not yet undergone significant melting, did not move. This can probably be explained by a high viscosity of partially molten fuel.

When the regular DEFORM option of a uniform cavity pressure is used, there is an inconsistency in the DEFORM/PLUTO2 transition due to the lack of an axial in-pin fuel motion model in DEFORM. The nodal pressures calculated in PLUTO2 which use the pressure-generating free fission gas and the nodal volume obtained from DEFORM can be different from the average cavity pressure. Since the latter is considered more realistic for all but very high power situations, the free fission-gas content of the PLUTO2 nodes is adjusted to give the DEFORM calculated average cavity pressure. When the axially nonuniform pressure option in DEFORM is chosen (only relevant for very high power situations), the PLUTO2-calculated nodal pressure can still be somewhat different from the one calculated by DEFORM and has to be slightly adjusted. This is because DEFORM takes into account the radial temperature, fuel density, and porosity profiles in the molten fuel cavity, whereas PLUTO2 uses a radially averaged temperature, fuel density and porosity in the cavity nodes. The above-mentioned inconsistencies in the transition from DEFORM to PLUTO2 are not considered to be serious. To remove them would require the inclusion of pre-failure in-pin fuel motion in DEFORM and the accounting of the radial temperature profile in the molten cavity in PLUTO2 or LEVITATE.

The initialization of the PLUTO2 and LEVITATE cavity is done in subroutines PLINPT (**PL** UTO2 **IN**PU**T**) and PLSET (**PL**UTO2 **SET**UP). DEFORM calculated cavity dimensions and free and dissolved fission-gas densities are transferred to PLUTO2 in these routines. Also, the radially averaged cavity temperatures are determined from the temperature profiles obtained from the heat-transfer routines, TSHTRN or TSHTRV.

14.2.3 Coupling of the In-pin Motion Calculation with the PLHTR Heat Transfer Calculation

For the calculation of the growth of the molten fuel cavity, it is important that a fuel temperature calculation is performed in the solid fuel surrounding the molten fuel cavity. This temperature calculation is performed with the PLHTR (PLUTO2 HEAT TRANSFER) routine which is a modified version of the transient heat-transfer subroutine TSHTRV. (The latter is used to calculate fuel-pin temperatures during the sodium boiling phase of a LOF accident). In the PLHTR heat conduction calculation, the fuel node adjacent to the molten cavity boundary receives heat from the molten cavity in the form of a heat source. This heat source is the total energy convected to the cavity wall from the molten fuel flow during a heat-transfer time step divided by the heat-transfer time-step size. If a solid fuel node reaches the input melt fraction criterion FNMELT, this node will be gradually added to the molten cavity. The rate of fuel addition depends on the temperature gradient between the node adjacent to the cavity wall and its neighboring solid fuel node.

14.2.4 Overview of the Numerical Approach for the In-pin Fuel Motion Calculation and Description of Subroutines PLIPIN and PL2PIN

There are several requirements for the solution algorithm for this one dimensional, compressible flow problem with variable flow cross section: (a) it had to be able to handle large mass sinks (due to fuel ejection from the pins), (b) it had to conserve mass perfectly, and (c) it had to run efficiently. This was achieved with a predominantly explicit Eulerian solution method. All convection mass, energy, and momentum fluxes are treated explicitly (i.e., the beginning of time-step values are used). However, the solution sequence of the different equations introduces a certain implicitness which is of importance for treating the strong mass sinks.

For the in-pin motion calculation three mass, one energy and one momentum conservation equations are solved. The mass conservation equations are for the fuel and the free and the dissolved fission gas. The fuel and free fission-gas mass conservation equations are solved first, followed by the fuel energy conservation equation. The fission-gas temperature change is assumed to be the same as the fuel temperature change. From the results of the mass and energy conservation equations, a new pressure is calculated. This is not the true end-of-time-step pressure because the velocity changes during the time step have not been included. However, it is a proper prediction for the end-of-time- step value in an explicit sense. In the fuel-ejecting nodes, this new pressure and the advanced densities and energies are used for calculating the fuel and fission-gas ejection rates. The pressure in the ejection nodes is then decreased in order to account for the fuel and gas losses. The adjusted new pressures are then used in the fuel/fission-gas momentum equation. There is an input option in PLUTO2 to use a combination of the new and the old pressures. This option lets the pressure, P, be:

$$P = (1 - \text{EPCH}) \cdot \text{beginning-of-time-step pressure} + \text{EPCH} \cdot \text{advanced pressure} \tag{14.2-1}$$

The recommended input value is EPCH $= 1$ because the calculation remained stable for longer time steps in test problems involving shock propagation and shock reflection when this input value was used [14-21]. In these test calculations, it was also attempted to iterate on the convective fluxes by repeating most of the in-pin calculation for each time step. This seemed to make the results somewhat smoother, but did not allow a time step as long as the one allowable with the above-described explicit scheme.

The time-step criterion in this compressible calculation is the sonic Courant condition. This does not require particularly small time steps because the sonic velocity in two-phase mixtures is fairly low for the void fractions encountered in pin blowdown calculations.

A staggered numerical grid with the densities, energies, and temperatures defined at the cell centers and the velocities at the cell edges is used. The spatial differencing uses full donor cell differencing. Although this is not as accurate as higher order differencing, it makes the calculation stable because it introduces a numerical diffusion effect. It is interesting to note that this stabilizing effect was not enough to keep two-phase test calculations stable for shock reflections from a rigid wall. Although these test calculations could be stabilized by introducing an artificial viscous pressure into the calculations [14-21], the artificial viscous pressure is not necessary for regular pin blowdown calculations. It could even cause problems if the input constants, which affect the magnitude of the artificial viscous pressure, are set too high. The two relevant input parameters, *C1VIPR* and *C2VIPR*, should therefore be set to small values such as 10^{-3}.

The free fission-gas mass conservation, the fuel mass conservation, and the fuel energy equation are solved in subroutine PLIPIN (*PL*UTO2 *1*st *PIN* ROUTINE). PLIPIN also computes the molten cavity geometry changes due to fuel melt-in, and the fuel and fission-gas ejection from the pins (see Section 14.3). The fuel/fission-gas momentum equation and the dissolved fission-gas mass equation are solved in PL2PIN (*PL*UTO2 *2*nd *PIN* ROUTINE). This routine also calculates the sonic velocities for each node. The minimum time step found is the predicted time step for the next cycle. However, the actual PLUTO2 time step, which is used both in the pin and in the channel calculation, is the smaller of the pin and channel hydrodynamics time steps. The channel time step is usually the smaller one, and thus, dominates the time-step selection.

14.2.5 Definition of the Generalized Smear Densities for the In-pin Calculation

The use of generalized smear densities in PLUTO2 (and LEVITATE) has been prompted by the many different moving and stationary components in this problem. Its use also simplifies the differential and finite difference equations for variable cross section flow. The pie chart in Figure 14.2.1 gives an example of the relative cross sectional areas within a subassembly or experimental loop at a certain axial elevation.

If the total area of the subassembly is AXMX, the generalized volume fraction of a certain component k is:

$$\theta_{\mathrm{k}}(z,t) = \frac{A_{\mathrm{k}}(z,t)}{\mathrm{AXMX}} \tag{14.2-2}$$

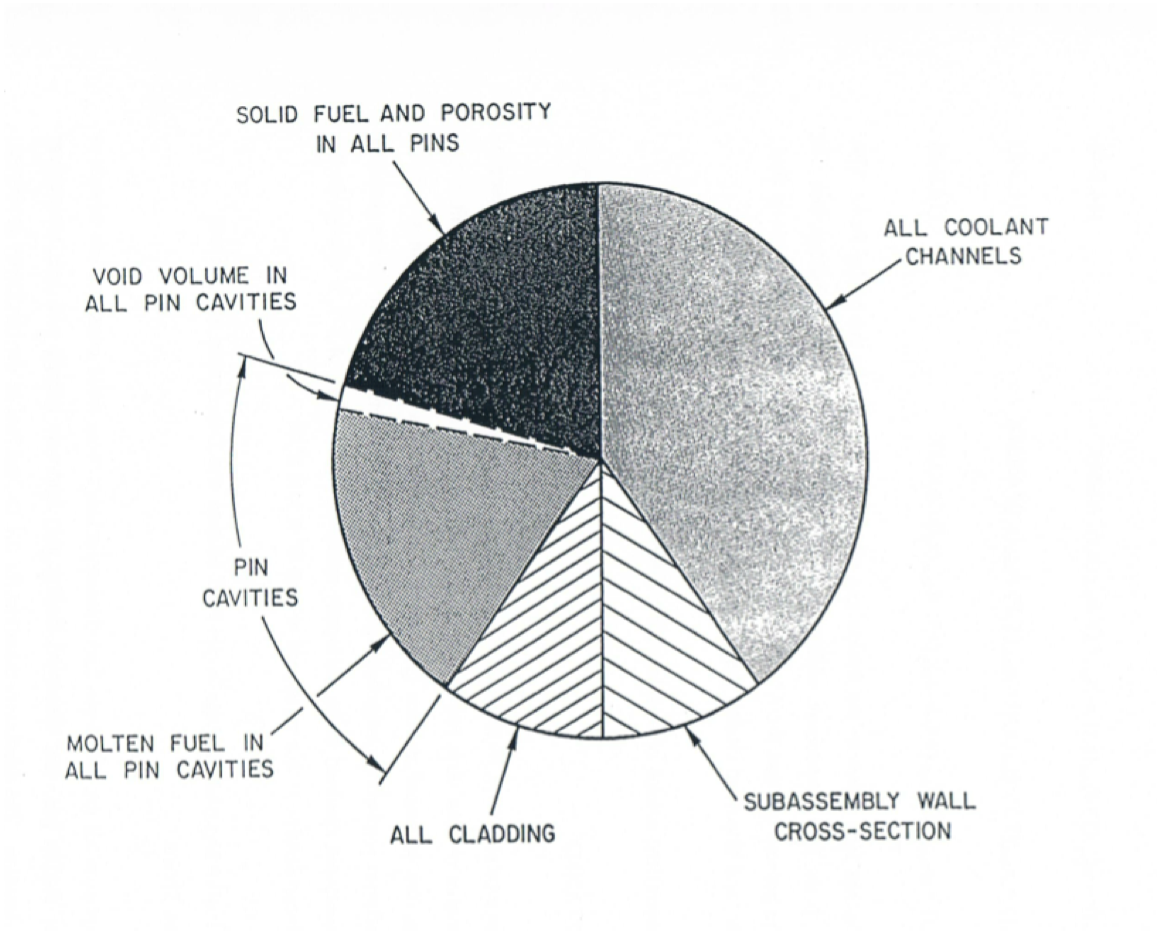


Figure 14.2.1: The Left-hand Side of This Pie Chart Illustrates the Possible Material Cross Sectional Areas in the Fuel Pin, the Whole Pie Representing an Area AXMW Which is an Input Parameter

where A_{k} is the cross sectional area occupied by component k (the latter refers, e.g., to the cavities or the moving fuel in all failed pins within the subassembly cross section). The reference area AXMX, which is an input quantity, can be arbitrarily chosen. This is because the code is invariant to the choice of AXMX (i.e., as long as it is not varied by several orders of magnitude which can lead to differences due to truncation errors). However, the recommended value of AXMX is the cross sectional area of a subassembly or experiment test

section (including the can wall) because the volume fractions of the different components that appear in the PLUTO2 output are better understood in this case.

The generalized volume fraction of the free fission gas and fuel vapor in the cavity is the difference between the cavity volume fraction and the fuel volume fraction.

$$\theta_{\text{fica}}(z,t) = \theta_{\text{ca}}(z,t) - \theta_{\text{fuca}}(z,t) \tag{14.2-3}$$

where

θ_{fica} = generalized volume fraction of the free fission gas and fuel vapor in the cavities of the failed pins.

θ_{ca} = generalized volume fraction of the molten cavities in all failed pins which can be calculated from $A_{\text{ca}}/\text{AXMX}$

θ_{fuca} = generalized volume fraction of the fuel in the cavities of all failed pins

Generalized smear densities, which are always marked by a prime, are defined as products of physical densities and generalized volume fractions:

$$\rho'_{\text{fuca}}(z,t) = \rho_{\text{fuca}}(T)\,\theta_{\text{fuca}}(z,t) = \frac{\rho_{\text{fuca}}(T)\,A_{\text{fuca}}(z,t)}{\text{AXMX}} \tag{14.2-4}$$

$$\rho'_{\text{fica}}(z,t) = \rho_{\text{fica}}(T,P)\,\theta_{\text{fica}}(z,t) = \frac{\rho_{\text{fica}}(T,P)\,A_{\text{fica}}(z,t)}{\text{AXMX}} \tag{14.2-5}$$

$$\rho'_{\text{fsca}}(z,t) = \rho_{\text{fsca}}(z,t)\,\theta_{\text{fuca}}(z,t) = \frac{\rho_{\text{fsca}}(z,t)\,A_{\text{fuca}}(z,t)}{\text{AXMX}} \tag{14.2-6}$$

where the subscript fsca refers to the fission gas which is dissolved in the cavity fuel. The temperature T is the fuel temperature, which should actually be written with subscript fuca. It should again be pointed out that the A's refer to total cross section areas of all the cavity fuel, free fission gas, and dissolved fission gas in the failed pins of one subassembly.

The generalized source or sink term is written as:

$$S' = \frac{S^l}{\text{AXMX}} \tag{14.2-7}$$

where the source or sink term S^l represents a mass source or sink per unit time and unit length. The primed source or sink term has the dimension of mass per unit time and per unit smear volume. This unit of smear volume is a m^3 of the cell volume $\text{AXMX} \cdot \Delta z$ in which all relevant components (including components in all failed pins, the components in all the channels and the structure) are assumed to be uniformly smeared. From Eq. (14.2-7), a change in the generalized smear density ρ' due to source or sink S' it just $S'\Delta t$.

14.2.6 Differential Equations for the In-pin Fuel Motion and Description of Sink and Source Terms

The equation set for the in-pin fuel motion includes three mass conservation equations (for fuel, free fission gas, and dissolved fission gas) and one energy and one momentum conservation equation. The continuity equation for the molten fuel in the pin cavity is written:

$$\frac{\partial}{\partial \text{t}}(\rho_{\text{fuca}} A_{\text{fuca}}) = -\frac{\partial}{\partial \text{z}}\left(\rho_{\text{fuca}} A_{\text{fuca}} u_{\text{fuca}}\right) + S^l_{\text{fuca,me}}(z,t) - S^l_{\text{fuca,ej}}(z,t) \tag{14.2-8}$$

where the subscripts me and ej refer to fuel melting into and fuel ejection from the pin cavities of all failed pins, respectively. By dividing by AXMX and using the definitions of the generalized smear densities and source and sink terms, Eq. (14.2-8) becomes:

$$\frac{\partial}{\partial \text{t}}\rho'_{\text{fuca}} = -\frac{\partial}{\partial \text{z}}\left(\rho'_{\text{fuca}} u_{\text{fuca}}\right) + S'_{\text{fuca,me}} - S'_{\text{fuca,ej}} \tag{14.2-9}$$

where the primed sources and sinks are per unit time and per unit smear volume (see Eq. (14.2-7)). The integrated source term, $S'_{\text{fuca,me}}\Delta t_{\text{PL}}$, which is actually needed in the finite difference equations of the code, is calculated from

$$S'_{\text{fuca,me}}\Delta t_{\text{PL}} = \rho_{\text{fu,cabd}}\Delta A_{\text{me}}\text{NRPI}\cdot\frac{\text{FNPI}}{\text{AXMX}} \tag{14.2-10}$$

here

Δt_{PL} = PLUTO2 time step

$\rho_{\text{fu,cabd}}$ = fuel density (including porosity) adjacent to the cavity boundary,

ΔA_{me} = area of fuel (including porosity) melted into the cavity per PLUTO2 time step per pin

NRPI = number of pins per subassembly,

FNPI = fraction of the pins which are failed.

When the solid fuel node adjacent to the cavity has not yet exceeded an input melt fraction value FNMELT:

$$S'_{\text{fuca,me}}\Delta t_{\text{PL}} = 0 \tag{14.2-11}$$

The ΔA_{me} in Eq. (14.2-10) is related to a change in the cavity diameter by

$$\Delta A_{\text{me}} = \frac{\pi}{4}\Delta D_{\text{ca}}\left(2D_{\text{ca}} + \Delta D_{\text{ca}}\right) \tag{14.2-12}$$

In order to avoid adding the whole radial node instantaneously upon meeting the input melt fraction criterion FNMELT, the radial node is added gradually beginning at the time FNMELT is reached and the addition is completed once the melt fraction has exceeded FNMELT $+ 0.1$. Once the melt fraction has become greater than FNMELT, the change in cavity diameter is calculated from

$$\Delta D_{\text{ca}} = \text{FNMECA}\cdot 2\cdot\Delta r_{\text{node}} \tag{14.2-13}$$

where Δr_{node} is the width of the heat-transfer node adjacent to the cavity wall before it is melted into the cavity. FNMECA is the fraction of this node width that has melted into the cavity per PLUTO2 time step.

$$\text{FNMECA} = \frac{\Delta t_{\text{PL}}\left(\frac{\left(T^{n+1}_{\text{cabd}} - T^{n}_{\text{cabd}}\right)}{\Delta t_{\text{Ht}}}\right)}{\left(0.1\cdot\left(T_{\text{fu,liq}} - T_{\text{fu,sol}}\right)\right)} = \frac{\Delta T_{\text{PL}}}{\left(0.1\cdot\left(T_{\text{fu,liq}} - T_{\text{fu,sol}}\right)\right)} \tag{14.2-14}$$

where

$T^{n}_{\text{cabd}}, T^{n+1}_{\text{cabd}}$ are the temperatures of the fuel node adjacent to the cavity at the beginning and at the end of the heat-transfer time step Δt_{Ht}, respectively.

$T_{\text{fu,liq}}, T_{\text{fu,sol}}$ are fuel liquidus and solidus temperatures, respectively. The difference between the two is the melting band width.

Δt_{PL} is the PLUTO2 time step.

ΔT_{PL} is the temperature change of the fuel adjacent to the cavity during a PLUTO2 time step.

Eq. (14.2-14) implies that the whole heat-transfer node will be melted into the cavity after its temperature has risen by 1/10 of the melting band width beyond the input value FNMELT. Moreover, it is checked whether the neighboring solid fuel node also has exceeded the input melt fraction criterion FNMELT. If this is the case, the entire remaining node currently melting into the cavity is added immediately to the cavity. This situation can occur in TREAT experiments.

The sink term $S'_{\text{fuca,ej}}$ that is due to fuel ejection is also the source term for the coolant channel equations. It will be described in detail in Section 14.3. Its integrated form is similar to Eq. (14.2-10)

$$S'_{\text{fuca,ej}} \cdot \Delta t_{\text{PL}} = \rho'_{\text{fuca}} \cdot \text{FN} \tag{14.2-15}$$

where

FN = fraction of the fuel in an ejecting cavity cell that is actually ejected during a PLUTO2 time step.

The free fission-gas mass conservation equation is written

$$\frac{\partial}{\partial \text{t}} \left(\rho_{\text{fica}} A_{\text{fica}} \right) = -\frac{\partial}{\partial \text{z}} \left(\rho_{\text{fica}} A_{\text{fica}} u_{\text{fuca}} \right) + S^l_{\text{fica,me}} \left(z, t \right) - S^l_{\text{fica,ej}} + S^l_{\text{fsca,rl}} \tag{14.2-16}$$

where the subscripts fsca and rl refer to fission gas in solution and to the release of this dissolved fission gas, respectively. Dividing by AXMX and by using the definition of the generalized smear densities and source and sink terms yields

$$\frac{\partial}{\partial \text{t}} \rho'_{\text{fica}} = -\frac{\partial}{\partial \text{z}} \left(\rho'_{\text{fica}} u_{\text{fuca}} \right) + S^l_{\text{fica,me}} - S^l_{\text{fica,ej}} + S^l_{\text{fsca,rl}} \tag{14.2-17}$$

The integrated source term for free fission gas due to fuel melt-in is similar to that in Eq. (14.2-10):

$$S'_{\text{fica,me}} \Delta t_{\text{PL}} = \rho_{\text{fifs,cabd}} \Delta A_{\text{me}} \left(\text{NRPI} \cdot \frac{\text{FNPI}}{\text{AXMX}} \right) \cdot \text{FIFNGB} \tag{14.2-18}$$

where ΔA_{me} is the calculated change in cavity cross sectional area described before (see Eq. (14.2-12)), $\rho_{\text{fifs,cabd}}$ is the density of all the gas in the fuel node adjacent to the cavity, and FIFNGB is an input parameter determining the fraction of all the gas entering the cavity with the molten fuel which instantaneously becomes free gas. This is an input parameter because DEFORM does not currently discriminate between intra-granular and grain-boundary gas. The ratio of the grain-boundary gas to the total gas in a node may be similar to the fraction of the gas that becomes free upon melting into the cavity. However, it seems more likely that only a smaller fraction will be instantaneously available because some of the grain-boundary bubbles may not be large enough to be considered free gas as soon as the fuel melts.

The quantity $\rho_{\text{fifs,cabd}}$ is calculated from:

$$\rho_{\text{fifs,cabd}} = \text{RETFG2}_{\text{cabd}} \frac{\left(\frac{\text{FUMS}_{\text{cabd}}}{\text{FUELM}} S_{\text{cabd}} \right)}{\text{VOLUME}_{\text{cabd}}} \tag{14.2-19}$$

where

$\text{RETFG2}_{\text{cabd}}$ is the total retained fission-gas mass in the original radial fuel-pin node at the cavity boundary before it has actually melted in,

$\text{FUMS}_{\text{cabd}}$ is is the current fuel mass in the melting radial fuel-pin node at the cavity boundary,

$\text{FUELMS}_{\text{cabd}}$ is the original fuel mass of the radial fuel-pin node at the cavity boundary before any fuel is removed due to melt-in,

$\text{VOLUME}_{\text{cabd}}$ is the current volume of the melting radial fuel-pin node at the cavity boundary.

The free fission-gas sink term due to fuel ejection in Eq. (14.2-17) is described in detail in Section 14.3. Its form is similar to that of the sink term for fuel ejection Eq. (14.2-15):

$$S'_{\text{fica,ej}} \cdot \Delta t_{\text{PL}} = \rho'_{\text{fica}} \cdot \text{FN} \tag{14.2-20}$$

The term FN used in this equation is the same as in Eq. (14.2-15) because it is assumed in PLUTO2 that fission gas and fuel are ejected with the same volume fractions that exist in the ejecting cavity node.

The source term due to dissolved fission-gas release in Eq. (14.2-17) is:

$$S'_{\text{fsca,r}l} = \rho'_{\text{fsca}} \cdot \text{CIRTFS} \tag{14.2-21}$$

where CIRTFS is a release constant for dissolved fission gas which is input and has the dimensions s^{-1}. (The same release constant is also used for the dissolved fission-gas release in the coolant channels - see Eq. (14.4-20)). This is a relatively simple exponential decay-type approach to treat the release of the gas dissolved in molten fuel. However, the understanding of the mechanism of dissolved gas release from molten fuel is still very limited.

The dissolved fission-gas mass conservation equation is:

$$\frac{\partial}{\partial \text{t}} \left(\rho_{\text{fsca}} A_{\text{fuca}}\right) = -\frac{\partial}{\partial \text{z}} \left(\rho_{\text{fsca}} A_{\text{fuca}} u_{\text{fuca}}\right) + S^l_{\text{fsca,me}} - S^l_{\text{fsca,ej}} - S^l_{\text{fsca,rl}} \tag{14.2-22}$$

By dividing this equation by AXMX and using the definitions of the generalized smear density and sources and sinks, one obtains:

$$\frac{\partial}{\partial \text{t}} \rho'_{\text{fsca}} = -\frac{\partial}{\partial \text{z}} \left(\rho'_{\text{fica}} u_{\text{fuca}}\right) + S^l_{\text{fsca,me}} - S^l_{\text{fsca,ej}} - S^l_{\text{fsca,rl}} \tag{14.2-23}$$

where

$$S'_{\text{fsca,me}} = S'_{\text{fica,me}} \frac{(1 - \text{FIFNGB})}{\text{FIFNGB}} \tag{14.2-24}$$

and $S'_{\text{fica,me}}$ and FIFNGB are defined for Eq. (14.2-18). The ejection term is connected to the one for the free fission-gas ejection, which is given in Eq. (14.2-20):

$$S'_{\text{fsca,ej}} = S'_{\text{fica,ej}} \frac{\rho'_{\text{fsca}}}{\rho'_{\text{fica}}} \tag{14.2-25}$$

The absolute value of the sink term due to the dissolved fission-gas release has been described in Eq. (14.2-21).

The fuel energy conservation equation is written:

$$\begin{aligned} \frac{\partial}{\partial \text{t}} \left(\rho_{\text{fuca}} e_{\text{fuca}} A_{\text{fuca}}\right) = &- \frac{\partial}{\partial \text{z}} \left(\rho_{\text{fuca}} e_{\text{fuca}} A_{\text{fuca}} u_{\text{fuca}}\right) + S^l_{\text{fuca,me}} e_{\text{fu,cabd}} \\ &- S'_{\text{fuca,ej}} e_{\text{fuca}} + Q A_{\text{fuca}} \rho_{\text{fuca}} \\ &- h_{\text{fuca,cabd}} \cdot \left(T_{\text{fuca}} - T_{\text{fu,cabd}}\right) \pi D_{\text{ca}} \text{NRPI} \cdot \text{FNPI} \end{aligned} \tag{14.2-26}$$

Dividing Eq. (14.2-26) by AXMX and using the definitions of the generalized smear densities and sources and sinks produces:

$$\begin{aligned} \frac{\partial}{\partial \text{t}} \left(\rho'_{\text{fuca}} e_{\text{fuca}}\right) = &- \frac{\partial}{\partial \text{z}} \left(\rho'_{\text{fuca}} e_{\text{fuca}} u_{\text{fuca}}\right) + S'_{\text{fuca,me}} e_{\text{fu,cabd}} \\ &- S^l_{\text{fuca,ej}} e_{\text{fuca}} + Q \rho'_{\text{fuca}} \\ &- h_{\text{fuca,cabd}} \cdot \left(T_{\text{fuca}} - T_{\text{fu,cabd}}\right) \pi D_{\text{ca}} \text{NRPI} \cdot \frac{\text{FNPI}}{\text{AXMX}} \end{aligned} \tag{14.2-27}$$

By rewriting the left-hand side of Eq. (14.2-27) as

$$\frac{\partial}{\partial \text{t}} \left(\rho'_{\text{fuca}} e_{\text{fuca}}\right) = \rho'_{\text{fuca}} \frac{\partial}{\partial \text{t}} e_{\text{fuca}} + e_{\text{fuca}} \frac{\partial}{\partial \text{t}} \rho'_{\text{fuca}} \tag{14.2-28}$$

and by using the mass conservation Eq. (14.2-8), one obtains

$$\begin{aligned} \rho'_{\text{fuca}} \frac{\partial}{\partial \text{t}} e_{\text{fuca}} = &- \frac{\partial}{\partial \text{z}} \left(\rho'_{\text{fuca}} e_{\text{fuca}} u_{\text{fuca}}\right) + e_{\text{fuca}} \frac{\partial}{\partial \text{z}} \left(\rho'_{\text{fuca}} u_{\text{fuca}}\right) \\ &+ S'_{\text{fuca,ej}} \left(e_{\text{fu,cabd}} - e_{\text{fuca}}\right) + Q \rho'_{\text{fuca}} \\ &- h_{\text{fuca,cabd}} \left(T_{\text{fuca}} - T_{\text{fu,cabd}}\right) \pi D_{\text{ca}} \text{NRPI} \cdot \frac{\text{FNPI}}{\text{AXMX}} \end{aligned} \tag{14.2-29}$$

where Q is the fission heat source per kg of fuel

$$Q = \text{FNPOHE} \cdot \text{POW} \cdot \text{PSHAPE}\,(K) \cdot (1 - \text{GAMSS} - \text{GAMTNC} - \text{GAMTNE}) \cdot \frac{F_{\text{POWER}}}{\text{FUMASS}\,(K)} \tag{14.2-30}$$

In Eq. (14.2-30),

$$\text{FNPOHE} = \exp\left[\text{POWCOF}\,(1) + \Delta t\,(\text{POWCOF}\,(2) + \Delta t_{\text{x}}\text{POWCOF}\,(3))\right] \tag{14.2-31}$$

where Δt_{x} is the time between the current PLUTO2 time and the beginning of the current main (point kinetics) time step. The POWCOF's are the coefficients that are found by fitting an exponential function to the power levels at the end of the last three point kinetics time step.

POW = steady-state power in the peak axial fuel-pin segment (see SAS4A input description)

PSHAPE (K) = ratio of pin power at axial node K to POW (see SAS4A input description)

GAMSS, GAMTNC, GAMTNE = fractions of total power for the direct heating of structure, coolant, and cladding, respectively. (See SAS4A input description)

FUMASS (K) = initial total fuel mass in axial pin segment K

F_{POWER} = power reduction factor if there is a radial power gradient in the pin (as is common in TREAT experiments):

$$F_{\text{POWER}} = \frac{\sum_{\text{I=1}}^{I_{\text{cabd}}} \text{PSHAPR}\,(I) \cdot \text{FUELMS}\,(I,K) \,/\, \sum_{\text{I=1}}^{I_{\text{cabd}}} \text{FUELMS}\,(I,K)}{\sum_{\text{I=1}}^{\text{NT}} \text{PSHAPR}\,(I) \cdot \text{FUELMS}\,(I,K) \,/\, \sum_{\text{I=1}}^{\text{NT}} \text{FUELMS}\,(I,K)} \tag{14.2-32}$$

where

NT = number of radial pin nodes

I_{cabd} = number of radial pin nodes in the cavity

PSHAPR (I) = mass normalized radial power distribution in radial node I (See SAS4A input description)

FUELMS (I,K) = initial fuel mass in the radial fuel-pin node I, K.

The heat-transfer coefficient in Eq. (14.2-26) is the sum of a convective and a conduction heat transfer term.

$$h_{\text{fuca,cabd}} = (h_1 + h_2)$$

where

$$h_1 = \frac{k_{\text{fu}}}{D_{\text{ca}}} \cdot \text{St} \cdot \text{Pr} \cdot \text{Re} \tag{14.2-33}$$

This comes from the definition of the Nusselt number

$$\text{Nu} = \text{St} \cdot \text{Pr} \cdot \text{Re} \tag{14.2-34}$$

where

St = Stanton number = $\frac{h}{(\rho u C_{\text{p}})}$

Pr = Prandtl number = $\mu \frac{C_{\text{p}}}{k}$

Re = Reynolds numbers = $D \frac{\rho u}{\mu}$

The Deissler correlation [14-22, 14-23] was used for finding the relationship between the three nondimensional numbers on the right-hand side of Eq. (14.2-34). The Prandtl number for fuel is about 2.2. By using this value in the Deissler correlation, it can be shown that

$$\mathrm{St} \approx 0.0158\mathrm{Re}^{-0.2} \tag{14.2-35}$$

By using Eq. (14.2-8) and the definition of the Prandtl number,

$$h_1 = \frac{1}{D_{\mathrm{ca}}} \cdot \mu_{\mathrm{fu,liq}} \cdot C_{\mathrm{p,fu}} \cdot \mathrm{CIA3} \cdot \mathrm{Re}_{\mathrm{ca}}^{0.8} \tag{14.2-36}$$

where

k_{fu} = conductivity of fuel which is input (CDFU)

$\mu_{\mathrm{fu,liq}}$ = liquid fuel viscosity which is input (VIFULQ)

CIA3 = input constant. A value of 0.0158 is recommended because of

$$\mathrm{Re}_{\mathrm{ca}} = \frac{\left(\frac{|u_{\mathrm{fuca}}| D_{\mathrm{ca}} \rho'_{\mathrm{fuca}}}{\theta_{\mathrm{ca}}}\right)}{\mu_{\mathrm{fu,liq}}} \tag{14.2-37}$$

The conduction heat-transfer coefficient, h_2, which is relevant for a low flow or a stagnant flow condition (the stagnant condition is assumed in the simplified PLUTO2 node — see Section 14.1.2), is of the following form

$$h_2 = \frac{k_{\mathrm{fu}}}{D_{\mathrm{ca}}} 4 \tag{14.2-38}$$

The pressure calculation in the fuel-pin cavity is based on the assumption that the fission-gas pressure and fuel-vapor pressure can be added. The total cavity pressure is:

$$P_{\mathrm{ca}} = P_{\mathrm{fica}}\left(T_{\mathrm{fuca}}, \rho_{\mathrm{fica}}\right) + P_{\mathrm{fvca}}\left(T_{\mathrm{fuca}}\right) \tag{14.2-39}$$

where the fission-gas pressure is calculated form a special form of the ideal-gas equation which takes the compressibility of the liquid fuel into account:

$$P_{\mathrm{fica}} = \frac{R_{\mathrm{fi}} \cdot T_{\mathrm{fuca}} \rho'_{\mathrm{fica}}}{\left(\theta_{\mathrm{ca}} - \theta_{\mathrm{fuca}} + \theta_{\mathrm{fuca}} K_{\mathrm{fu}} \cdot P_{\mathrm{fica}}\right)} \tag{14.2-40}$$

where

R_{fi} = the universal gas constant divided by the molecular weight of the mixture of fission gas and helium fill gas. R_{fi} is equal to an input value RGAS which should take into account the relative amounts of krypton, xenon, and helium. The latter may be important for near-fresh fuel.

K_{fu} = liquid fuel compressibility which is input (see CMFU).

The physically reasonable solution of the quadratic Eq. (14.2-40) is:

$$P_{\mathrm{fica}} = \frac{-\left(\theta_{\mathrm{ca}} - \theta_{\mathrm{fuca}}\right) + \sqrt{\left(\theta_{\mathrm{ca}} - \theta_{\mathrm{fuca}}\right)^2 + 4\theta_{\mathrm{fuca}} K_{\mathrm{fu}} \cdot R_{\mathrm{fi}} \cdot T_{\mathrm{fuca}} \rho'_{\mathrm{fica}}}}{2\theta_{\mathrm{fuca}} K_{\mathrm{fu}}} \tag{14.2-41}$$

There is also a second solution with a minus sign in front of the square root, which does not give a physically reasonable result. Eq. (14.2-41) reduces to a single-phase liquid pressure solution for no fission gas and $\theta_{\mathrm{fuca}} > \theta_{\mathrm{ca}}$:

$$P_{\mathrm{fica}} = -\frac{\left(\theta_{\mathrm{ca}} - \theta_{\mathrm{fuca}}\right)}{\left(\theta_{\mathrm{fuca}} K_{\mathrm{fu}}\right)} \tag{14.2-42}$$

which is equivalent to the definition of the fuel compressibility. For void fractions greater than 30% the fission-gas pressure is calculated from a modified Eq. (14.2-40) in which the term with K_{fu} is dropped.

If the dissolved fission gas present in the molten fuel cavity is assumed to occupy a negligible volume, in comparison with the volume occupied by the free gas then the θ_{fuca} in Eq. (14.2-40) and 14.2-34 reflects the actual volume occupied by the molten fuel. This assumption is justified when the ambient cavity pressure is significantly smaller than the bubble surface tension pressure, as was the case in all TREAT experiments analyzed with SAS4A. When this assumption is used, the dissolved fission gas does not affect the cavity pressure until it is released, in a gradual manner, to the free gas field.

However, during a transient characterized by a rapid increase in the power level, the cavity pressures can reach higher values, so that the influence of the dissolved fission gas becomes important. In order to address this issue the dissolved fission gas modeling in the pin cavity was introduced as described below.

The dissolved fission gas is assumed to exist in the form of small bubbles, with the gas inside the bubbles at the pressure P_{bubble}:

$P_{\text{bubble}} = P_{\text{surface tension}} + P_{\text{cavity}}$.

The quantity $P_{\text{surface tension}}$ is an input to the code (PRSFTN). Its value can be determined as:

$$P_{\text{surface tension}} = \frac{2\sigma}{r},$$

where:

σ = bubble surface tension,

r = bubble radius.

The recommended value for $P_{\text{surface tension}}$ is 4×10^7 Pa, corresponding to $\sigma = 0.4$ N/m (400 dyne/cm) and $r = 2 \times 10^{-8}$ m (200 Å). The surface tension pressure is assumed to remain constant during the calculations. Furthermore, in order to simplify the calculations, the pressure P_{cavity} used in the calculation of the bubble pressure P_{bubble} is the cavity pressure at the end of the previous time step. The volume occupied by the dissolved gas in the axial cavity cell i is then calculated using the equation:

$$V_{\text{D,i}} = \frac{M_{\text{D,i}} \cdot R \cdot T_{\text{i}}}{m \cdot P_{\text{bubble,i}}} = \frac{R}{m} \cdot \frac{M_{\text{D,i}} \cdot T_{\text{i}}}{P_{\text{bubble,i}}}$$

where:

$M_{\text{D,I}}$ = mass of dissolved fission gas in cell i,

$\frac{R}{m}$ = universal gas constant divided by molar mass,

T_{i} = gas temperature in cell i, assumed to be equal to the molten fuel temperature,

$P_{\text{bubble,I}}$ = the pressure of the gas contained in the small bubbles, calculated above.

The fuel and dissolved gas are assumed to form a homologous mixture, with the volume given by:

$$V_{\text{fuel}}^{\text{mixture}} = V_{\text{fuel,i}} + V_{\text{D,i}},$$

where:

$V_{\text{fuel,I}}$ = molten fuel volume in cell i,

$V_{\text{D,i}}$ = dissolved gas volume in cell i.

The calculation then proceeds as before, using the volume $V_{\text{fuel,i}}^{\text{mixture}}$ instead of $V_{\text{fuel,i}}$ to determine the value of θ_{fuca} used in Eq. (14.2-40) and Eq. (14.2-41).

When there is little molten fuel in an axial cavity cell, i.e., the gas occupies a large fraction of the cell volume, a perfect gas equation is used to determine the free gas pressure. The underlying assumption is that the compressibility of the fuel/dissolved-gas-mixture has a negligible effect under these circumstances. However, if larger amounts of molten fuel are present, the pressure calculation takes into account the molten fuel compressibility. In this case, the dissolved fission gas and molten fuel volumes are lumped together and assumed to have the same compressibility as the molten fuel. This is a simplifying assumption, but is conservative in the sense that the molten fuel/dissolved gas mixture is likely to have a higher compressibility than the molten fuel itself and thus would lead to somewhat lower cavity pressures during rapid power excursions.

This treatment of the dissolved fission gas has also been implemented in DEFORM-4, LEVITATE, and PLUTO2. If the input constant PRSFTN is less than 10^7 the effect of the dissolved fission gas on the ambient pressure is ignored.

The momentum conservation equation for the fuel/fission-gas mixture is:

$$\begin{aligned}
\frac{\partial}{\partial \mathrm{t}}\left(\rho_{\mathrm{fuca}} A_{\mathrm{fuca}} u_{\mathrm{fuca}} + \rho_{\mathrm{fica}} A_{\mathrm{fica}} u_{\mathrm{fuca}}\right) &= -\frac{\partial}{\partial \mathrm{z}}\left(\rho_{\mathrm{fuca}} A_{\mathrm{fuca}} u_{\mathrm{fuca}}^2 + \rho_{\mathrm{fica}} A_{\mathrm{fica}} u_{\mathrm{fuca}}^2\right) \\
-A_{\mathrm{ca}} \cdot \left(\frac{\partial \mathrm{P}_{\mathrm{ca}}}{\partial \mathrm{z}} + \frac{\partial \mathrm{P}_{\mathrm{vi}}}{\partial \mathrm{z}}\right) - g\left(\rho_{\mathrm{fuca}} A_{\mathrm{fuca}} + \rho_{\mathrm{fica}} A_{\mathrm{fica}}\right) & \\
-u_{\mathrm{fuca}}\left|u_{\mathrm{fuca}}\right| \cdot \left(\rho_{\mathrm{fuca}} \frac{A_{\mathrm{fuca}}}{A_{\mathrm{ca}}} + \rho_{\mathrm{fica}} \frac{A_{\mathrm{fica}}}{A_{\mathrm{ca}}}\right) \frac{A_{\mathrm{ca}} \cdot F_{\mathrm{friction}}}{(2D_{\mathrm{ca}})} & \\
- \left(S^l_{\mathrm{fuca,ej}} + S^l_{\mathrm{fica,ej}}\right) u_{\mathrm{fuca}} &
\end{aligned} \tag{14.2-43}$$

Dividing Eq. (14.2-43) by AXMX and using the definitions for the generalized smear densities and mass sinks gives:

$$\begin{aligned}
\frac{\partial}{\partial \mathrm{t}}\left[u_{\mathrm{fuca}}\left(\rho'_{\mathrm{fuca}} + \rho'_{\mathrm{fica}}\right)\right] &= \frac{\partial}{\partial \mathrm{z}}\left[u_{\mathrm{fuca}}^2\left(\rho'_{\mathrm{fuca}} + \rho'_{\mathrm{fica}}\right)\right] \\
-\theta_{\mathrm{ca}}\left(\frac{\partial \mathrm{P}_{\mathrm{ca}}}{\partial \mathrm{z}} + \frac{\partial \mathrm{P}_{\mathrm{vi}}}{\partial \mathrm{z}}\right) - g\left(\rho'_{\mathrm{fuca}} + \rho'_{\mathrm{fica}}\right) & \\
-u_{\mathrm{fuca}}\left|u_{\mathrm{fuca}}\right| \cdot \left(\rho'_{\mathrm{fuca}} + \rho'_{\mathrm{fica}}\right) \frac{F_{\mathrm{friction}}}{(2D_{\mathrm{ca}})} & \\
- \left(S'_{\mathrm{fuca,ej}} + S'_{\mathrm{fica,ej}}\right) u_{\mathrm{fuca}} &
\end{aligned} \tag{14.2-44}$$

u_{fuca} = vertically upward fuel velocity in the cavity (see the sign of the gravity head term in Eq. (14.2-43)).

P_{vi} = artificial viscous pressure for stabilizing test problems involving shock wave propagation and shock reflection from a rigid boundary. Not necessary for regular cavity blowdown problems.

If the net free fission-gas mass flux into a numerical node is negative (i.e., for net outflow), then $P_{\mathrm{vi}} = 0$.

If the net convective free fission-gas mass flux into a numerical cell is positive [14-27]:

$$P_{\mathrm{vi}} = \frac{\mathrm{C2VIPR} \cdot 0.5 \cdot \left(p'_{\mathrm{fu,K}} + p'_{\mathrm{fi,K}}\right) \cdot \left(u_{\mathrm{fuca,K+1}} - u_{\mathrm{fica,K}}\right)^2}{\theta_{\mathrm{ficaxx,K}}} \tag{14.2-45}$$

where

K = index of the axial cavity node for which P_{vi} is being evaluated

C2VIPR = input constant which determines the magnitude of the numerical damping.

$$\theta_{\mathrm{ficaxx}} = \begin{cases} \theta_{\mathrm{fica}} \text{ for } \theta_{\mathrm{fica}} > \theta_{\mathrm{ca}} \cdot \mathrm{C1VIPR} \\ \theta_{\mathrm{ca}} - \mathrm{C1VIPR} \text{ for } \theta_{\mathrm{fica}} < \theta_{\mathrm{ca}} \cdot \mathrm{C1VIPR} \end{cases} \tag{14.2-46}$$

where

C1VIPR = Input constant

The Moody friction factor F_{friction} in Eq. (14.2-44) depends on the Reynolds number

$$\text{Re}_{\text{pi}} = \frac{|u_{\text{fuca}}|\, D_{\text{ca}} \cdot (\rho'_{\text{fuca}} + \rho'_{\text{fica}})}{(\theta_{\text{cs}} \cdot \text{VIFULQ})} \tag{14.2-47}$$

where VIFULQ is the viscosity of liquid fuel which is input. The friction factor is

$$F_{\text{friction}} = \begin{cases} \frac{64}{\text{Re}_{\text{pi}}} & \text{for Re}_{\text{pi}} \text{ CIREFU} \\ \text{CIFRFU} & \text{for Re}_{\text{pi}} \text{ CIREFU} \end{cases} \tag{14.2-48}$$

where CIREFU and CIFRFU are both input and should be made consistent in order to avoid a jump in the friction factor at $\text{Re}_{\text{pi}} = \text{CIREFU}$.

The last term in Eq. (14.2-44) is necessary because the total momentum in the cell decreases due to ejection and this has to be considered in a momentum equation written in conservative form. There is no term accounting for the fuel melt-in because this fuel is added to the cavity with zero axial velocity, and therefore does not change the total momentum. However, since the generalized fuel smear density will change in a cell with melt-in, this will lead to a velocity decrease in such a cell.

14.2.7 Finite Difference Equations for the In-pin Motion

In the overview of the numerical scheme given in Section 14.2.4, it was pointed out that full donor cell spatial differencing and a largely explicit time differencing are used for treating the in-pin motion. The implicit aspect of the solution is that the mass and energy conservation equations are solved first and then a pressure is calculated on the basis of the mass and energy equation results. This advanced pressure is used in the momentum conservation equation.

The finite differencing of all the mass conservation equations is the same. The fuel mass conservation is used as an illustration.

$$\frac{\left(\rho'^{n+1}_{\text{fuca,K}} - \rho'^{n}_{\text{fuca,K}}\right)}{\Delta t_{\text{PL}}} = -\frac{\left((\rho' u)_{\text{fuca,K+1}} - (\rho' u)_{\text{fuca,K}}\right)}{\Delta z_{\text{K}}} + \sum_{\text{k}} S'_{\text{K,k}} \tag{14.2-49}$$

where

$$(\rho' u)_{\text{fuca,K}} = \begin{cases} \rho'_{\text{fuca,K-1}} u_{\text{K}} & \text{for } u_{\text{K}} > 0 \\ \rho'_{\text{fuca,K}} u_{\text{K}} & \text{for } u_{\text{K}} < 0 \end{cases} \tag{14.2-50}$$

u_{K} = fuel velocity at the mesh cell boundary K.

$\rho'_{\text{fuca,K-1}}$ = generalized fuel smear density at the mesh cell midpoint below boundary K

$\rho'_{\text{fuca,K}}$ = generalized fuel smear density at the mesh cell midpoint above boundary K

The numerical grid used in the program was discussed in Section 14.2.4, which gives an overview of the numerical scheme. Densities are at the cell centers and velocities at the cell edges (see schematic before the momentum conservation Eq. (14.2-61)). The source and sink terms have already been described in their finite difference form in the previous section.

The convective fluxes at the lower and upper boundaries of the cavity that are located in the end cells KK1 and KKMX, respectively, are

$$(\rho' u)_{\text{fuca,KK1}} = 0 \tag{14.2-51}$$

and

$$(\rho' u)_{\text{fuca,KKMX+1}} = 0 \tag{14.2-52}$$

The end cells are not always the same during a PLUTO2 run because the molten cavity can extend axially. When a new cell is added to the cavity, mass can only flow into this end cell if its cross section is at least 20% of that of the neighboring cell in the molten cavity. This had to be done because of problems with overcompression of cells with a very small cross section.

The finite difference form of the fuel energy Eq. (14.2-29) is:

$$\begin{aligned} \frac{\rho'_{\text{fuca,K}}\left(e^{n+1}_{\text{fuca,K}} - e^{n}_{\text{fuca,K}}\right)}{\Delta t_{\text{PL}}} &= -\left[\left(\rho'_{\text{fuca}} e_{\text{fuca}} u_{\text{fuca}}\right)^{n}_{\text{K+1}}\right. \\ &\left. - \left(\rho'_{\text{fuca}} e_{\text{fuca}} u_{\text{fuca}}\right)^{n}_{\text{K}}\right] / \Delta z_{\text{K}} + e^{n}_{\text{fuca,K}} \left[\left(\rho'_{\text{fuca}} u_{\text{fuca}}\right)^{n}_{\text{K+1}}\right. \\ &\left. - \left(\rho'_{\text{fuca}} u_{\text{fuca}}\right)^{n}_{\text{K}}\right] / \Delta z_{\text{K}} + S'^{n}_{\text{fuca,me,K}} \left(e^{n}_{\text{fu,cabd,K}} - e^{n}_{\text{fuca,K}}\right) \\ + Q^{n}_{\text{K}} \rho'^{n}_{\text{fuca,K}} - h^{n}_{\text{fuca,cabd,K}} \left(T^{n}_{\text{fuca,K}} - T^{n}_{\text{fu,cabd,K}}\right) \pi D^{n}_{\text{ca,K}} &\frac{\text{NRPI} \cdot \text{FNPI}}{\text{AXMX}} \end{aligned} \tag{14.2-53}$$

The fuel cavity temperature T_{fuca} in the above equation has to be calculated from the internal energy:

$$\begin{aligned} &\text{For } e_{\text{fu,sol}} < e_{\text{fuca}} < e_{\text{fu,liq}} \ , \\ &T_{\text{fuca}} = T_{\text{fu,sol}} + \left(T_{\text{fu,liq}} - T_{\text{fu,sol}}\right) \cdot \frac{\left(e_{\text{fuca}} - e_{\text{fu,sol}}\right)}{\left(e_{\text{fu,liq}} - e_{\text{fu,sol}}\right)} \end{aligned} \tag{14.2-54}$$

$$\begin{aligned} &\text{For } e_{\text{fuca}} > e_{\text{fu,liq}} \ , \\ &T_{\text{fuca}} = T_{\text{fu,liq}} + \frac{\left(e_{\text{fuca}} - e_{\text{fu,liq}}\right)}{C_{\text{p,fu}}} \end{aligned} \tag{14.2-55}$$

where $C_{\text{p,fu}}$ is the fuel specific heat which is the single input value CPFU. The convective energy flux at cell boundary K in Eq. (14.2-53) in calculated as:

$$\left(\rho'_{\text{fuca}} e_{\text{fuca}} u_{\text{fuca}}\right)_{\text{k}} = \begin{cases} \left(\rho'_{\text{fuca}} e_{\text{fuca}}\right)_{\text{K-1}} u_{\text{fuca,K}} \text{for } u_{\text{fuca,K}} > 0 \\ \left(\rho'_{\text{fuca}} e_{\text{fuca}}\right)_{\text{K}} u_{\text{fuca,K}} \text{for } u_{\text{fuca,K}} < 0 \end{cases} \tag{14.2-56}$$

At the lower and upper cavity ends that are in cells KK1 and KKMX, the convective energy fluxes are zero.

$$\left(\rho'_{\text{fuca}} e_{\text{fuca}} u_{\text{fuca}}\right)^{n}_{\text{KK1}} = 0 \tag{14.2-57}$$

and

$$\left(\rho'_{\text{fuca}} e_{\text{fuca}} u_{\text{fuca}}\right)^{n}_{\text{KKMX+1}} = 0 \tag{14.2-58}$$

By using the convective fluxes from Eq. (14.2-50) and Eq. (14.2-56), and the definitions of the energy gain and loss terms given earlier, and by differencing the second term of Eq. (14.2-53) like the first, Eq. (14.2-53) can be solved for $e^{n+1}_{\text{fuca,K}}$. Fuel temperatures that are shown in the PLUTO2 output are calculated by using Eq. (14.2-54) and Eq. (14.2-55).

For the finite difference form of the momentum equation the following quantities have to be defined at the edges of the numerical cells: the combined fuel/fission-gas generalized smear density and the cavity volume fraction. These quantities become:

$$\rho'_{\text{fufi,bk}} = 0.5\left(\rho'_{\text{fuca,K-1}} + \rho'_{\text{fuca,K}}\right) + 0.5\left(\rho'_{\text{fica,K-1}} + \rho'_{\text{fica,K}}\right) \tag{14.2-59}$$

$$\theta_{\text{ca,bk}} = 0.5\left(\theta_{\text{ca,K}} + \theta_{\text{ca,K-1}}\right) \tag{14.2-60}$$

where the subscript bk indicates that these quantities are at the lower boundaries of cell K. This is shown on the schematic below. On the numerical grid the velocities are also defined on the cell boundaries, whereas the pressures, densities and volume fractions are defined at the cell centers. This is also shown in the following schematic.

θ_{bk}	$\theta_{\text{bk+1}}$	
ρ'_{bk}	$\rho_{\text{bk+1}}$	
u_{K}	$u_{\text{K+1}}$	
$\theta_{\text{K-1}}$	θ_{K}	$\theta_{\text{K+1}}$

$\rho_{\text{K-1}}$	ρ_{K}	$\rho_{\text{K+1}}$
$P_{\text{K-1}}$	P_{K}	$P_{\text{K+1}}$
$S_{\text{K-1}}$	S'_{K}	$S'_{\text{K+1}}$

z_{K}	$z_{\text{K+1}}$

The finite difference form of the momentum conservation Eq. (14.2-44), is written using the definitions 14.2-46 and 14.2-47 as:

$$\begin{aligned}\frac{\left(\rho'^{n+1}_{\text{fufi,bk}}u^{n+1}_{\text{fuca,K}}-\rho'^{n}_{\text{fufi,bk}}u^{n}_{\text{fuca,K}}\right)}{\Delta t_{\text{PL}}} &= -\Big[\left(u^2_{\text{fuca}}\rho'_{\text{fufi}}\right)^n_{\text{K}}\\ -\left(u^2_{\text{fuca}}\rho'_{\text{fufi}}\right)^n_{\text{K-1}}\Big]/\Delta z &- \theta_{\text{ca,bk}}\left[(1-\varepsilon)\cdot\left(P^n_{\text{ca,K}}-P^n_{\text{ca,K-1}}\right)\right.\\ \left.+\varepsilon\cdot\left(P^{n+1}_{\text{ca,K}}-P^{n+1}_{\text{ca,K-1}}\right)+\left(P^n_{\text{vi,K}}-P^n_{\text{vi,K-1}}\right)\right]/\Delta z &- g\rho'_{\text{fufi,bk}}\\ -u^{n+1}_{\text{fuca,K}}\left|u^n_{\text{fuca,K}}\right|\rho'^{n}_{\text{fufi,bk}}F_{\text{friction,bk}}/\left(2D_{\text{ca}}\right) &- \Big(S'^{n+1/2}_{\text{fuca,ej,K-1}}\\ +S'^{n+1/2}_{\text{fuca,ej,K}}+S'^{n+1/2}_{\text{fica,ej,K-1}}+S'^{n+1/2}_{\text{fica,ej,K}}\Big)\cdot\left(u^{n+1}_{\text{fuca,K}}+u^n_{\text{fuca,K}}\right)\cdot 0.25\end{aligned} \tag{14.2-61}$$

where

ε = Input value EPCH that can be between zero and one (see Eq. (14.2-1)),

and Δz implies $0.5\left(\Delta z_{\text{K-1}}+\Delta z_{\text{K}}\right)$.

By defining

$$S'^{n+1/2}_{\text{fuca,ej,bk}}=\left(S'^{n+1/2}_{\text{fica,ej,K-1}}+S'^{n+1/2}_{\text{fica,ej,K}}+S'^{n+1/2}_{\text{fica,ej,K-1}}+S'^{n+1/2}_{\text{fica,ej,K}}\right)\cdot 0.5 \tag{14.2-62}$$

and by collecting all terms with u_{uica}^{n+1} on the left-hand side of the equation, one obtains

$$\begin{aligned} u_{\text{fuca,K}}^{n+1}\Bigg[\frac{\rho'^{\,n+1}_{\text{fufi,bk}}}{\Delta t_{\text{PL}}}+\left|u_{\text{fuca,K}}^{n}\right|\rho'^{\,n}_{\text{fufi,bk}}\frac{F_{\text{friction,bk}}}{(2D_{\text{ca}})} \\ +S'^{n+1/2}_{\text{fufi,ej,bk}}\cdot 0.5\Bigg]=\frac{\rho'^{\,n}_{\text{fufi,bk}}u_{\text{fuca,K}}^{n}}{\Delta t_{\text{PL}}} \\ -\frac{\left[\left(u_{\text{fuca}}^{2}\rho'_{\text{fufi}}\right)_{\text{K}}^{n}-\left(u_{\text{fuca}}^{2}\rho'_{\text{fufi}}\right)_{\text{K-1}}^{n}\right]}{\Delta z} \\ -\theta_{\text{ca,bk}}\left[(1-\varepsilon)\cdot\left(P_{\text{ca,K}}^{n}-P_{\text{ca,K-1}}^{n}\right)\right. \\ \left.+\varepsilon\left(P_{\text{ca,K}}^{n+1}-P_{\text{ca,K-1}}^{n+1}\right)+\left(P_{\text{vi,K}}^{n}-P_{\text{vi,K-1}}^{n}\right)\right]/\Delta z-g\rho'_{\text{fufi,bk}} \\ -S'^{n+1/2}_{\text{fuca,ej,bk}}u_{\text{fuca,K}}^{n}\cdot 0.5 \end{aligned} \tag{14.2-63}$$

The convective momentum flux in Eq. (14.2-63) is calculated as

$$\left(u_{\text{fuca}}^{2}\rho'_{\text{fufi}}\right)_{\text{K}}=\begin{cases}\rho'_{\text{fufi,K}}u_{\text{fuca,K}}^{2} & \text{if } (u_{\text{fuca,K}}+u_{\text{fuca,K+1}})>0\\ \rho'_{\text{fufi,K}}u_{\text{fuca,K+1}}^{2} & \text{if } (u_{\text{fuca,K}}+u_{\text{fuca,K+1}})<0\end{cases} \tag{14.2-64}$$

$$\left(u_{\text{fuca}}^{2}\rho'_{\text{fufi}}\right)_{\text{K-1}}=\begin{cases}\rho'_{\text{fufi,K-1}}u_{\text{fuca,K-1}}^{2} & \text{if } (u_{\text{fuca,K-1}}+u_{\text{fuca,K}})>0\\ \rho'_{\text{fufi,K-1}}u_{\text{fuca,K}}^{2} & \text{if } (u_{\text{fuca,K-1}}+u_{\text{fuca,K}})<0\end{cases} \tag{14.2-65}$$

where

$$\rho'_{\text{fufi,K}}=\rho'_{\text{fuca,K}}+\rho'_{\text{fica,K}} \tag{14.2-66}$$

The momentum fluxes for the lower and upper end cells of the cavity, which are designated by KK1 and KKMX, are:

$$\left(u_{\text{fuca}}^{2}\rho'_{\text{fufi}}\right)_{\text{KK1}}=\begin{cases}\rho'_{\text{fufi,KK1}}u_{\text{fuca,KK1+1}}^{2}\cdot 0.25 & \text{if } u_{\text{fuca,KK1+1}}>0\\ \rho'_{\text{fufi,KK1}}u_{\text{fuca,KK+1}}^{2} & \text{if } u_{\text{fuca,KK1+1}}<0\end{cases} \tag{14.2-67}$$

$$\left(u_{\text{fuca}}^{2}\rho'_{\text{fufi}}\right)_{\text{KKMX}}=\begin{cases}\rho_{\text{fufi,KKMX}}u_{\text{fuca,KKMX}}^{2} & \text{if } u_{\text{fuca,KKMX}}>0\\ \rho_{\text{fufi,KKMX}}u_{\text{fuca,KKMX}}^{2}\cdot 0.25 & \text{if } u_{\text{fuca,KKMX}}<0\end{cases} \tag{14.2-68}$$

The factor 0.25 in the above convection terms comes from the assumption of a zero velocity at the end of the cavity.

The momentum Eq. (14.2-63) can be solved for $u_{\text{fuca,K}}^{n+1}$ if Eq. (14.2-59), Eq. (14.2-60), and Eq. (14.2-64) through Eq. (14.2-68) are used.

14.2.8 Time-step Determination for the In-Pin Motion

The PLUTO2 time step Δt_{PL} used in the numerical solution of all the in-pin and channel conservation equations is restricted by the sonic Courant conditions for both the in-pin and channel flows. The sonic Courant condition for the channel flow and the determination of the PLUTO2 time step Δt_{PL} is given at the end of the channel flow description in Section 14.4.6.4. In the present section, only the restriction imposed by the sonic Courant condition for the in-pin flow is described.

The time-step size for the in-pin motion is computed to be a fraction, 0.4, of the minimum time step based on the sonic Courant condition

$$\Delta t_{\text{PL,pin}}=0.4\cdot\min\left[\frac{z_{\text{K}}}{v_{\text{sonic,K}}+\left|u_{\text{fuca,K}}\right|}\right]_{\text{K = KK1,KKMX}} \tag{14.2-69}$$

The minimum in Eq. (14.2-69) is evaluated over all the axial cells of the molten fuel cavity. The sonic velocity is calculated from an expression for an adiabatic, homogeneous two-component gas-liquid mixture which is based on Eq. 27 in Reference [14-28].

$$v_{\text{sonic}}^2 = \gamma_{\text{fi}} P_{\text{fica}} \Big/ \left\{ \left[\alpha_{\text{fica}}^2 \cdot \rho_{\text{fica}} + \alpha_{\text{fica}} \left(1 - \alpha_{\text{fica}}\right) \rho_{\text{fuca}} \right] + \left[\left(1 - \alpha_{\text{fica}}\right)^2 \rho_{\text{fica}} + \alpha_{\text{fica}} \left(1 - \alpha_{\text{fica}}\right) \rho_{\text{fica}} \right] \gamma_{\text{fi}} P_{\text{fica}} K_{\text{fu}} \right\} \tag{14.2-70}$$

where

$\alpha_{\text{fica}} = \frac{\theta_{\text{fica}}}{\theta_{\text{ca}}}$ = void fraction in the cavity

$\gamma_{\text{fi}} = \frac{C_{\text{p,fi}}}{C_{\text{v,fi}}} = 1.4$ (value assumed in PLUTO2)

$K_{\text{fu}} =$ CMFU = input liquid fuel compressibility

The above equation holds for adiabatic gas behavior, although the in-pin fission-gas treatment in PLUTO2 is isothermal (the gas temperature is assumed to be always equal to the fuel temperature). However, the sonic velocity for adiabatic gas behavior is higher than that for isothermal gas behavior, and thus, leads to a more conservative (i.e., smaller) time step. Moreover, if a pure gas flow were treated in sections of pins with a prefabricated central hole, the current time-step determination would actually be necessary.

14.3 Fuel and Fission-gas Ejection from the Pins

14.3.1 Physical Model and Assumptions

In the previous section, the sinks for the in-pin motion due to fuel and fission-gas ejection appeared in the fuel, free fission gas, and dissolved fission-gas mass conservation equations (see Eq. (14.2-9), 14.2-15, and 14.2-20). These sink terms are, of course, source terms in the channel thermal-hydraulics treatment.

The approach taken for the calculation of the fuel and fission-gas ejection from the pins is based on the assumption that there will always be pressure equilibrium established between a cavity node and the adjacent channel node if fuel and fission-gas ejection occurs. This assumption is justified for cladding ruptures with dimensions larger than one-pin diameter because of the very short distance between cavity nodes and corresponding coolant channel nodes. For pin-hole type failures leading to a pure gas release, which is not currently treated in PLUTO2, an orifice equation for calculating this gas release may be more appropriate. In the older PLUTO code [14-1, 14-3, 14-4], both an asymptotic orifice equation and an ejection calculation, based on pressure equilibration, were computed for the fuel and gas ejections, and the smaller of the two predictions was used. The asymptotic orifice equation usually predicted the higher ejection rates. These were causing a higher pressure in the channel node than in the adjacent cavity node which was considered non-physical.

To achieve the pressure equilibrium between the cavity and channel nodes in PLUTO2, an estimated amount of fuel and gas (with the same volume ratio as present in the cavity node) is ejected into the channel and the resulting cavity and channel pressures are calculated. This calculation is repeated several times with updated estimates for the ejected fuel and fission-gas masses until a pressure equilibrium is achieved. This approach does not consider (or require) the cladding rupture area, which is an advantage because the evolution of the cladding rupture is not known. The cladding rupture sizes found in the post-test examinations of TREAT tests H5 [14-24] and J1 [14-25], which were terminated at a time when the pins were still largely intact, show large enough rupture sizes to justify the assumption of pressure equilibration between the cavity node behind the rupture and the adjacent channel node. Moreover, these rupture areas are considerably larger than the cross section of the molten cavity (50% areal melt fraction in an FFTF type pin means a molten cross sectional area of about 0.1 cm^2; cladding rupture areas in H5 and J1 are about 1 cm^2). This means that the controlling aspect of the fuel and gas ejection (at least in mild TOP accidents) is really the cross sectional area of the cavity, which is taken into account in the in-pin motion calculation in PLUTO2. In the SAS3D model SAS/FCI, which does not consider the in-pin motion

mechanistically, the cross sectional area for the orifice equation, used in the calculation of the SAS/FCI fuel and gas ejection, has to be set to about twice the cross sectional area of the cavity near the failure location [14-3]. Under LOF-driven-TOP conditions, when axial cladding rupture propagation is likely (see Section 14.3.3), the in-pin motion is not dominant in controlling the ejection because several contiguous cavity nodes may eject fuel and gas simultaneously. In addition, the time scales are very short under these conditions, leading to less axial fuel and gas convection into an ejection node than can actually be ejected.

Fuel and fission gas are being ejected from the fuel pins with the same volume ratio as exists in the ejecting cavity node. For conditions involving high void fractions in the ejecting cavity node, it is possible that fission gas or fuel vapor will be ejected preferentially because the remaining fuel may be in an annular flow configuration.

Another assumption about the fuel and gas ejection is that no backflow of materials from the channel to the pin is allowed when the pressure in the coolant channel node adjacent to the rupture becomes higher than the pressure in the ejection node. To model this backflow would be difficult and is not warranted because only a small amount of liquid sodium flowing back into the pin would lead to enough vaporization and pressurization to inhibit further backflow of materials. However, the potential pressure increase in the cavity node may diminish the in-pin fuel motion. This could be beneficial if midplane failures are assumed. However, most channel pressurizations that have been observed experimentally and that may be due to FCIs are not sufficient in magnitude or duration to cause concern about this assumption.

14.3.2 Numerical Solution of the Fuel and Gas Ejection Calculation

The numerical evaluation of the amount of fuel and fission gas ejected during a PLUTO2 time step is performed as a second step after the fuel mass conservation Eq. (14.2-9) has been solved and advanced to the end of the time step without accounting for the ejection term $S'_{\text{fuca,ej}}$. This section describes how the fuel smear density value obtained at the end of the time step without accounting for ejection $\rho'^{\text{old}}_{\text{fuca}}$ is corrected for fuel ejection, thus obtaining the actual fuel smear density at the end of time step the $\rho'^{\text{new}}_{\text{fuca}}$ and the resulting equilibrated pressure of the ejecting cavity cell.

If FN_k is the fraction of the fuel in a cavity node that is ejected during a time step, the new generalized fuel smear density in the cavity node will be:

$$\rho'^{\text{new}}_{\text{fuca}} = \rho'^{\text{old}}_{\text{fuca}} - \rho'^{\text{old}}_{\text{fuca}} \cdot \text{FN}_\text{k} = \rho'^{\text{old}}_{\text{fuca}} - \text{FF} \tag{14.3-1}$$

where $\text{FF} = \rho'^{\text{old}}_{\text{fuca}} \cdot \text{FN}_\text{K}$ and is actually the same as the sink term for fuel ejection, $S'_{\text{fuca, ej}} \cdot \Delta t_{\text{PL}}$ (see Eq. (14.2-15)).

Since the fission gas and fuel are assumed to be ejected with the same volume ratio as present in the ejecting cell, the new generalized free-fission-gas smear density is

$$\rho'^{\text{new}}_{\text{fica}} = \rho'^{\text{old}}_{\text{fica}} - \rho'^{\text{old}}_{\text{fica}} \cdot \text{FN}_\text{k} = \rho'^{\text{old}}_{\text{fica}} - \text{FF} \cdot \frac{\rho'^{\text{old}}_{\text{fica}}}{\rho'^{\text{old}}_{\text{fuca}}} \tag{14.3-2}$$

The new cavity fuel fraction will be

$$\theta^{\text{new}}_{\text{fica}} = \theta^{\text{old}}_{\text{fica}} - \frac{\text{FF}}{\rho_{\text{fuca}}} \tag{14.3-3}$$

where $\rho_{\text{fuca}} = \rho_{\text{fuca}}(T_{\text{fuca}})$ is the theoretical fuel density.

By using Eq. (14.3-1), Eq. (14.3-2), and Eq. (14.3-3) in the cavity pressure calculation of Eq. (14.2-39) and Eq. (14.2-40) and by dropping the superscripts “old”, one obtains

$$\begin{aligned} P^{\text{new}}_{\text{ca}} = P_{\text{fvca}} + R_{\text{fi}} \cdot T_{\text{fuca}} \cdot \left(\rho'_{\text{fica}} - \text{FF} \cdot \frac{\rho'_{\text{fica}}}{\rho'_{\text{fuca}}} \right) \\ \Big/ \left\{ \theta_{\text{ca}} - \theta_{\text{fuca}} + \frac{\text{FF}}{\rho_{\text{fuca}}(T)} + \left[\theta_{\text{fuca}} - \frac{\text{FF}}{\rho_{\text{fuca}}(T)} \right] K_{\text{fu}} \cdot (P^{\text{new}}_{\text{ca}} - P_{\text{fuca}}) \right\} \end{aligned} \tag{14.3-4}$$

where

$\theta_{\text{ca}} - \theta_{\text{fuca}} = \theta_{\text{fica}}$

By collecting the terms with FF on the left-hand side one obtains

$$\begin{aligned} \text{FF} \cdot \Bigg[& \frac{(P_{\text{ca}}^{\text{new}} - P_{\text{fvca}})}{\rho_{\text{fuca}}} + \frac{\rho'_{\text{fica}}}{\rho'_{\text{fuca}}} \cdot R_{\text{fi}} \cdot T_{\text{fuca}} \\ & - \frac{K_{\text{fu}}}{\rho_{\text{fu}}} \cdot (P_{\text{ca}}^{\text{new}} - P_{\text{fvca}})^2 \Bigg] = R_{\text{fi}} \cdot T_{\text{fuca}} \cdot \rho'_{\text{fica}} \\ & - \theta_{\text{fica}} (P_{\text{ca}}^{\text{new}} - P_{\text{fvca}}) - \theta_{\text{fica}} \cdot K_{\text{fu}} \cdot (P_{\text{ca}}^{\text{new}} - P_{\text{fvca}})^2 \end{aligned} \tag{14.3-5}$$

This equation is used to calculate the fuel ejection FF for an estimated $P_{\text{ca}}^{\text{new}}$. FF is used, in turn, to update the fission-gas and fuel densities in the channel node adjacent to the ejecting cavity node. A new channel pressure is then calculated based on the updated fuel and gas densities. The method of calculating the channel pressures is discussed in Section 14.4.5. If the new channel pressure is not within 1% of the estimated new cavity pressure, a better estimate for the new cavity pressure is made and the calculational sequence for the fuel and gas ejection is repeated.

Because the conditions in the coolant channel node can vary from normal coolant flow to fully voided, the procedure of estimating the new cavity pressure is somewhat complex. It has evolved through trial and error and it usually leads to convergence in a few iterations. If the iteration does not converge, it usually indicates that a non-physical condition has developed in the pin or channel. When the axial extent of the cladding rupture includes more numerical nodes as a result of axial pin failure propagation (which is likely under LOF′d′TOP conditions), the ejection calculation will be performed for each of the axial failure nodes.

The ejection calculation from the pins was originally formulated to allow a simultaneous ejection from the three pin failure groups at any given axial location. As mentioned earlier, the second and third pin failure groups are not yet operational. Because of the complexity of the current ejection calculation from only one pin failure group, it is expected that a simultaneous ejection from three pin failure groups into the same coolant channel node may become too complicated. Therefore, a simpler approach, which would only allow the ejection from the cavity node with the highest pressure during a PLUTO2 time step, may have to be adopted. In this approach it would, of course, still be possible for the different pin failure groups to eject fuel simultaneously at different axial locations.

The fuel and gas ejection calculation is done at the end of subroutine PL1PIN which was discussed in Section 14.1.2. The ejection calculation is done in the calculational sequence after all in-pin as well as channel mass and energy conservation have been solved and before the momentum equations for both the in-pin and channel flows are solved. The ejection calculation thus provides updated pressures for the momentum equations.

14.3.3 Axial Pin Failure Propagation

Rapid axial pin failure propagation due to the stress concentrations at the edges of the initial rupture (zipper-type failure propagation) has not been accepted as a mechanism for rapid axial failure propagation (i.e., on a millisecond or submillisecond time scale) [14-26]. However, if the conditions in a fuel-pin node are such that failure of the cladding should occur irrespective of cladding stress concentrations due to an adjacent cladding rupture and whether the potential new failure node location is adjacent to the initial failure or not, there is no reason to disallow this type of failure propagation.

The failure of nodes other than the initial one is determined by PLUTO2 because the DEFORM calculation is not done in a calculational channel once PLUTO2 (or LEVITATE) has been initiated. There are two options for pin failure propagation in PLUTO2. The first option is a mechanistic one based on the burst strength of the cladding. The second option allows the pin failure propagation to proceed according to a combination of input failure criteria based on fuel melt fraction, cladding temperature, and pressure differential between cavity and channel.

The mechanistic option (KFAILP=0) assumes that the solid fuel surrounding the molten cavity is completely cracked and that the radial stress at the fuel-cladding interface can therefore be calculated by reducing the cavity pressure by a geometrical reduction factor.

$$\sigma_{\text{radial,fc,K}} = \frac{P_{\text{ca,K}} \cdot D_{\text{ca,K}}}{(2R_{\text{cl,fc,K}})} \tag{14.3-6}$$

where

$R_{\text{cl,fc,K}}$ is the inner radius of the cladding at axial node K.

$D_{\text{ca,K}}$ is the diameter of the molten cavity at axial node K.

The hoop stress in the cladding can be calculated from

$$\sigma_{\text{hoop,K}} = (\sigma_{\text{radial,fc,K}} - P_{\text{ch,i}}) \frac{R_{\text{cl,fc,K}}}{\Delta R_{\text{cl,K}}} \tag{14.3-7}$$

where $\Delta R_{\text{cl,K}}$ is the cladding thickness at axial node K, and subscript i refers to the channel node which is adjacent to cavity node K; $i = K +$ IDIFF where IDIFF is the number of channel nodes below cavity node $K = 1$.

By substituting Eq. (14.3-6) for the radial stress in Eq. (14.3-7),

$$\sigma_{\text{hoop,K}} = \frac{(P_{\text{ca,K}} \cdot D_{\text{ca,K}} - 2 \cdot P_{\text{ch,i}} \cdot R_{\text{cl,fc,K}})}{(2\Delta R_{\text{cl,K}})} \tag{14.3-8}$$

Failure is assumed to occur when the following two conditions are satisfied:

$$\sigma_{\text{hoop,K}} > \sigma_{\text{UTS}}\,(T_{\text{cl,in,K}}) \tag{14.3-9}$$

$$\text{AREA}_{\text{ca,K}} > \text{FNARME} \cdot \pi \cdot R^2_{\text{cl,fc,K}} \tag{14.3-10}$$

where

FNARME = an input melt fraction (or more precisely cavity cross sectional area divided by cross sectional area of the fuel) which should be set to ≥ 0.2 because the PLUTO2 in-point motion model requires a moderately sized cavity diameter to run well.

σ_{UTS} = the burst strength of the cladding that is a function of cladding temperature. This is calculated from the middle cladding node temperature in function subroutine UTS, which is described in Chapter 8 on fuel behavior.

A problem with this burst failure criterion is that it is not compatible with a melt fraction failure criterion, cladding strain failure criterion or any other criteria that are not directly related to the cladding strength. Since these latter criteria may be used in DEFORM to predict the initial failure, care has to be taken that the cladding will not rip open along sizable length as soon as PLUTO2 takes over the calculation. For example, this could happen if the initial pin failure was assumed to occur due to a melt fraction criterion at a time when the cavity pressure was already high enough to burst the cladding at several axial elevations. If a melt fraction criterion was chosen for the initial failure prediction, the problem that several nodes will instantaneously rupture once PLUTO2 initiates, can be avoided by setting the input parameter FNARME (see Eq. (14.3-10)) equal to or greater than the input melt fraction criterion for the initial failure.

The non-mechanistic input failure propagation criterion (KFAILP=1) involves several separate criteria. Failure occurs when the cladding middle node and outer surface temperatures, the cross sectional area of the cavity, and the pressure difference between cavity and channel exceed input criteria.

$$T_{\text{cl,in,K}} \quad \text{and} \quad T_{\text{cl,os,K}} > \text{TEFAIL} \tag{14.3-11}$$

and

$$\text{AREA}_{\text{ca,K}} > \text{FNARME} \cdot \pi \cdot R^2_{\text{cl,fc,K}} \tag{14.3-12}$$

and

$$P_{\text{ca,K}} > P_{\text{ch,i}} + \text{PRFAIL} \tag{14.3-13}$$

where

TEFAIL, FNARME, and PRFAIL are input parameters.

Subscripts **in** and **os** refer to the middle and outer cladding nodes, respectively. This combined criterion can of course be simplified by setting some of the input values to extreme numbers. For example, if TEFAIL is set to zero and PRFAIL to -10^{20}, this criterion will reduce to a pure melt fraction criterion. This input failure propagation criterion may be useful for comparison calculations with codes that allow failure propagation only according to one of the above non-mechanistic criteria. Moreover, if the initial failure criterion is a melt fraction criterion, one may also want to do the failure propagation based on a melt fraction criterion or on a melt fraction criterion combined with the condition that some overpressure has to exist in the molten cavity nodes.

14.3.4 Complete Pin Disruption and Switch to LEVITATE

Once the cladding has melted at a certain axial location and extensive fuel melting has also occurred at the same location, complete fuel-pin disruption will eventually occur. This cannot be treated with PLUTO2, and therefore, a switch to LEVITATE is made under these conditions. For each pin node, PLUTO2 checks whether the middle and outer cladding temperatures have exceeded the liquidus and whether the fuel-surface temperature has exceeded the solidus.

$T_{\text{cl, in, K}} > T_{\text{cl, liq}}$

and

$T_{\text{cl, os, K}} > T_{\text{cl, liq}}$

and

$T_{\text{fu, os, K}} > T_{\text{fu,sol}}$

where

cl, **in** refers to the middle cladding node,

cl, **os** refers to the outer cladding node,

fu, **os** refers to the outermost fuel node.

If the above conditions are met, control will be transferred from PLUTO2 to LEVITATE. These checks are performed at the end of the PLUTO2 driver routine. At that location in the code, it is also checked whether cladding melting has occurred in more than NCPLEV nodes (NCPLEV is input). If this criterion is met, a switch to LEVITATE will also occur. Moreover, a switch to LEVITATE is also made if the fuel temperature in the cavity or in the channel exceeds 4000 K because significant fuel vapor pressures will develop beyond this temperature.

14.4 Channel Hydrodynamics Model

14.4.1 Overview and Definition of Generalized Smear Densities

The treatment of the material motion in the coolant channels is more complex than that inside the pins because several material components (fuel, sodium, fission gas) and three different phases (solid, liquid, and gas) are involved. Of particular importance are the mass, energy and momentum interactions between the components and phases of this multi-component, multi-phase flow. These are largely determined by the local fuel configurations, which are in turn determined by the fuel flow regimes. Figure 14.4.1 depicts the possible fuel and liquid sodium configurations in an equivalent cylindrical channel. The outer perimeter of this equivalent cylindrical channel represents outer cladding surfaces and inner subassembly wall surfaces. The fuel flow regimes are extensively discussed in Section 14.4.3.1. Figure 14.4.2 gives an overview of the interactions between the various channel components that are described in detail in Section 14.4.3.4.

A one-dimensional two-fluid approach with variable flow cross section is used to calculate the momentum changes in the coolant channels. This means that two momentum conservation equations are solved. One of them calculates the velocity changes of the mixture of liquid sodium, sodium vapor, fission gas, and fuel vapor; the other calculates the velocity changes of the moving liquid or solid fuel. This approach is taken because the fuel-to-liquid sodium density ratio is more than 10, and a relatively rapid separation of these two components is likely. This is quite important for treating fuel coolant interactions (FCI's) because it may be one of the major limiting processes.

There are four mass conservation equations for tracking moving components in PLUTO2; (a) for liquid or solid fuel and for fuel vapor[3], (b) for liquid sodium and sodium vapor, (c) for free fission gas, and (d) for fission gas still dissolved in the fuel. There are also mass conservation equations for the stationary fuel crust (i.e., plated-out solid fuel) and for the stagnant sodium film.

A single energy conservation equation is solved for the mixture of liquid sodium (including the stagnant sodium film), sodium vapor, and free fission gas, which are all assumed to be in thermodynamic equilibrium. An energy conservation equation is solved for the moving solid fuel or for the two-phase mixture of liquid fuel and fuel vapor. The latter two are assumed to be in thermodynamic equilibrium in the region where the fuel vapor pressure is larger than 10^{-2}MPa. Outside this region, no fuel vapor is considered. There is also an energy equation for the solid fuel crusts.

The equation-of-state calculates the pressure for cells containing (a) liquid-phase sodium, (b) two-phase sodium, two-phase fuel and fission gas, and (c) superheated vapor and fission gas.

Although Eulerian hydrodynamics is generally employed in the formulation of the equations, a Lagrangian treatment is used to track the moving upper and lower boundaries of the multiphase region. All finite difference equations are written in conservative form. A staggered mesh grid, with the velocities at the cell edges, and pressures, densities, and temperatures at the cell centers, is used. The spatial derivatives are evaluated by full donor cell differencing, which is also known as second upwind differencing. This approach is more stable for very non-continuous conditions that are quite common in PLUTO2 applications although this is not as accurate as higher order schemes [14-30, 14-31].

The time differencing is mainly explicit, but there are also important implicit features involved. First, advantage is taken of the solution sequence of the conservation equations. The results of the mass conservation equations which are solved first, are used in the energy conservation equations. The results of these two sets of conservation equations and the results from the calculation of the fuel and gas ejection from the pins are used in the equation-of-state and provide an estimate of the channel pressure at the end of the time step. This is then used in the solution of the momentum conservation equations.

Another implicit feature that is of key importance for a stable velocity calculation of the lighter component is

[3] *Fuel vapor is not yet operational.

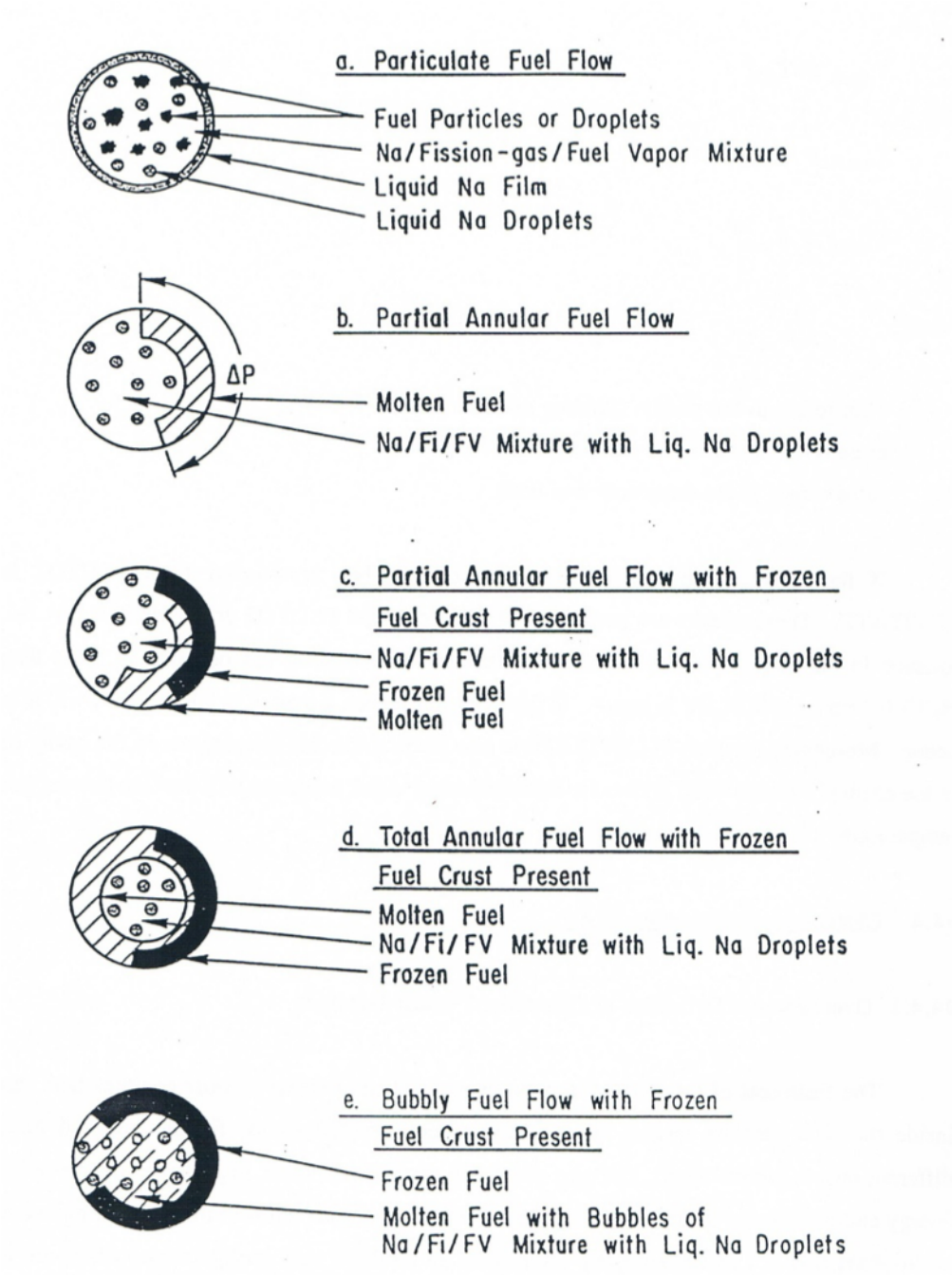


Figure 14.4.1: Illustration of the Possible Sodium and Fuel Flow Regimes as Well as the Fuel Freezing Patters in the PLUTO2 Model of the Coolant Channel

Components	Symbols	Moving or Stationary		fu	fs	Nℓ	Nv	fi	Fv*	cℓ	sr	ff	fsff	fm
Mobile Liquid or Solid Fuel	fu	Moving Components	UFCH			Ht Mo	Ht Mo	Ht Mo	Mo Ma	Ht Mo	Ht Mo	Ht Mo Ma		Ht
Dissolved gas in Mobile fuel	fs							Ma					Ma	
Moving Liquid Sodium	Nℓ,mv		UMCH	Ht Mo			Ma	Eq	Ht	Ht Mo	Ht Mo	Ht Mo		Ma
Sodium Vapor	Nv			Ht Mo		Ma		Eq	Ht	Ht Mo	Ht Mo	Ht Mo		Mo Ma
Free Fission gas	fi			Ht Mo	Ma	Eq	Eq			Ht Mo	Ht Mo	Ht Mo		Mo
Fuel Vapor*	fv			Mo Ma		Ht				Ht Mo	Ht Mo	Ht Mo		Ht Mo
Cladding	cℓ	Stationary		Ht Mo		Ht Mo	Ht Mo	Ht Mo	Ht Mo			Ht		Ht
Subassembly Wall	sr			Ht Mo Ma		Ht Mo	Ht Mo	Ht Mo	Ht Mo			Ht		Ht
Frozen Fuel Crusts	ff			Ht Mo Ma		Ht Mo	Ht Mo	Ht Mo	Ht Mo	Ht	Ht			
Dissolved fission Gas in crusts	fsff				Ma									
Liquid Sodium Film	fm			Ht		Ma	Mo Ma	Mo	Ht Mo	Ht	Ht			

*Not yet operational.

Legend

Ht = Heat transfer
Mo = Momentum transfer
Ma = Mass transfer
Eq = Thermal and Velocity Equilibrium

Figure 14.4.2: Table 14.4-1. Overview of the Interaction Between the Channel Components in PLUTO2

the simultaneous solution of the two momentum equations. The lighter component can experience strong and opposing pressure gradient and drag forces, which lead to oscillatory behavior for low density mixtures if a simultaneous solution is not performed [14-32]. Also important is the implicit treatment of the potentially large heat transfer between the various hot and cold components.

The explicit aspect of the solution method is that the conservation equations at different axial locations are not solved in a coupled fashion. This means, for example, that a new velocity calculated at one mesh cell boundary does not have an immediate effect on the new velocity at the neighboring boundary. By not solving the equation sets simultaneously, one can avoid three potential problems. First, an iterative solution for the inversion of a complicated matrix, which can have convergence problems or may require many iterations, can be avoided. (A direct solution through Gaussian elimination does not seem to be easily feasible for more than two coupled equation sets.) Second, the later addition of certain terms to the equation set is relatively straightforward for explicit schemes but can be complicated for implicit schemes. Terms may have to be added to the equation set in order to make a code such as PLUTO2 work for heretofore unforeseen conditions or because the importance of additional terms may have to be investigated. Third, the matrix elements of an implicit solution technique for multi-phase, multicomponent flows are usually complex and have no direct physical meaning. Therefore, they can complicate the debugging of such implicit solution schemes. However, a very stable implicit scheme that would decrease the code running time and diminish the truncation error introduced by the many time steps in an explicit scheme is desirable. This may not be a practical goal for the solution of the entire equation set in PLUTO2 but it would be a reasonable goal for selected equations such as the mass and momentum conservations of the light components.

Definition of the Generalized Smear Densities. A special feature of the PLUTO2 module (and of the LEVITATE module) is the use of generalized smear densities. This has been prompted by the presence of the many different components and the simplification of the differential and finite difference equations. The volume fractions are lumped together with the physical densities of the materials and therefore do not appear in the equations. Details about the smear densities in the fuel pin have already been described in Section 14.2.5. The pie chart in Figure 14.4.3 gives an example of the relative cross sectional areas involved at a certain axial location.

As discussed in Section 14.2.1, the same total cross sectional area of the pie, which is an input value AXMX, is used at all axial locations of the subassembly or the experimental loop considered. The relative sizes of the pie pieces can vary at different axial locations; pieces such as the pin piece can actually disappear at elevations above or below the pin bundles. There is no empty piece in the pie if one assumes the total area of the pie, AXMX, to be the total cross sectional area of a subassembly or an experimental loop. However, at elevations where the cross sectional area of the subassembly or the experimental loop is smaller than AXMX, an empty pie piece is necessary.

The calculational results of the PLUTO2 module should not depend on the choice of AXMX. This has been shown to be true as long as one does not change AXMX by too many orders of magnitude. If one changes AXMX very much, small differences in the results can occur due to differences in truncation errors. Thus, the choice of the input value of AXMX is one of convenience. The best choice for AXMX is usually the cross sectional area of the entire subassembly including the hexcan wall at an axial location where pins are present because this will lead to physically meaningful values of the volume fraction. If one uses an AXMX different from the above choice, the empty pie piece can become larger if a large AXMX is chosen, or even negative if a relatively small AXMX is chosen.

The definition of a volume fraction of a certain component k is:

$$\theta_{\mathrm{k}}(z,t) = \frac{A_{\mathrm{k}}(z,t)}{\mathrm{AXMX}} \tag{14.4-1}$$

where A_{k} is the actual cross sectional area occupied by component k (e.g., all the cladding or all the frozen fuel) inside a fuel assembly or inside the can wall of an experimental loop at axial location z and time t. If AXMX was chosen to be the entire subassembly cross section, the θ_{k} gives the actual volume fraction of component k in a slice of a subassembly or an experimental loop. For all axial elevations,

$$\theta_{\mathrm{pin}}(z) + \theta_{\mathrm{sr}}(z) + \theta_{\mathrm{ch}}(z) + \theta_{\mathrm{empty}}(z) = 1 \tag{14.4-2}$$

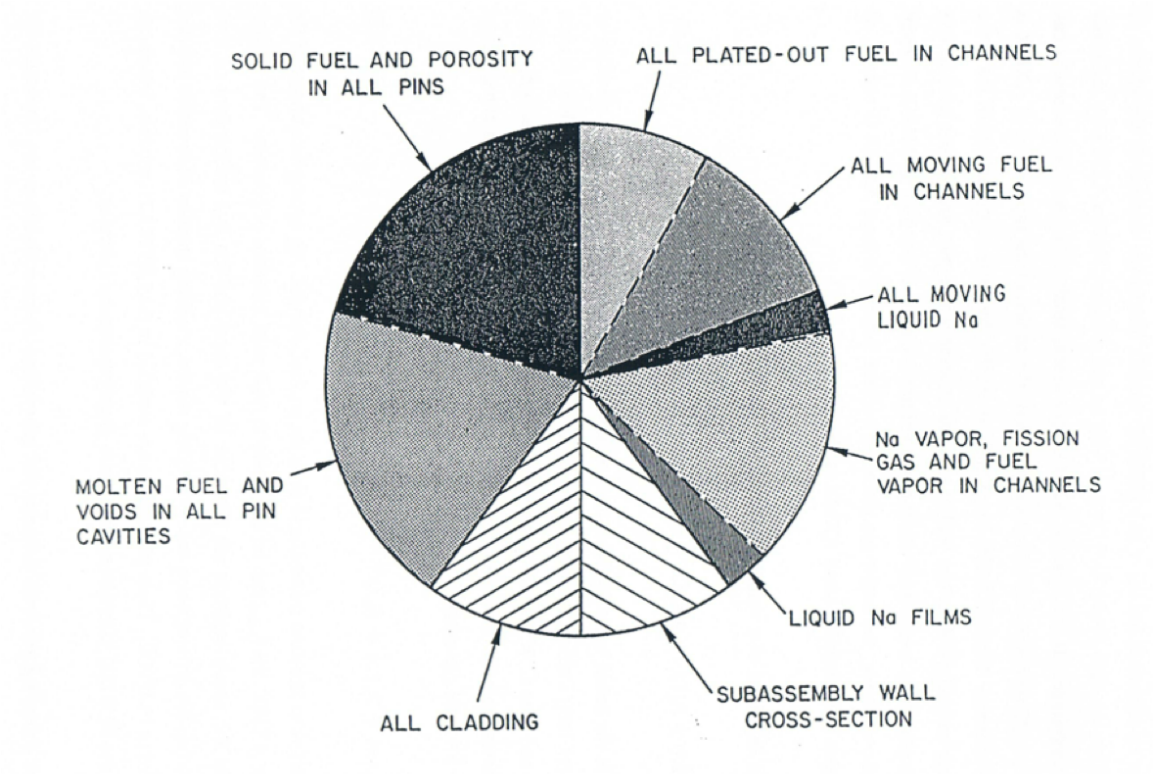


Figure 14.4.3: The Right-hand Side of This Pie Chart Illustrate the Possible Material Cross Sectional areas in the Coolant Channel, the Whole Pie Representing an Area AXMX, which is an Input Parameter

where **sr** refers to structure material and **ch** to the coolant channel. By appropriate choice of the input value AXMX, the θ_{empty} can be made zero at most axial locations of concern. The volume fractions in the above equation are functions of the axial position but independent of time because no fuel-pin disintegration and no cladding motion, cladding ablation or structure ablation are modeled in PLUTO2. These phenomena are modeled in the LEVITATE module, which can take over the PLUTO2 calculation when these phenomena come into play.

The following relations between the generalized volume fractions are important for writing the channel hydrodynamics equations in PLUTO2:

$$\theta_{\text{pin}}(z) = \theta_{\text{ch,op}}(z,t) + \theta_{\text{ff}}(z,t) \tag{14.4-3}$$

where

θ_{ch} is the generalized coolant channel volume fraction which is always equal to the values at pin failure in PLUTO2;

$\theta_{\text{ch, op}}$ is the generalized volume fraction of the open channel (i.e., the part of the channel available for the mobile components);

θ_{ff} is the generalized volume fraction of the frozen fuel crusts;

$$\theta_{\text{ch,op}}(z,t) = \theta_{\text{fu}}(z,t) + \theta_{N1}(z,t) + \theta_{\text{vg}}(z,t) \tag{14.4-4}$$

where

θ_{fu} is the generalized volume fraction of the mobile liquid or solid fuel in the channel;

θ_{N1} is the generalized volume fraction of the liquid sodium;

θ_{vg} is the generalized volume fraction of the vapor/gas (i.e., void) space in the channel;

$$\theta_{\text{N1}}(z,t) = \theta_{\text{N1,mv}}(z,t) + \theta_{\text{fm}}(z,t) \tag{14.4-5}$$

where

$\theta_{\text{N1, mv}}$ is the generalized volume fraction of the moving liquid sodium;

θ_{fm} is the generalized volume fraction of the liquid sodium film.

The following generalized smear densities are defined:

$$\begin{aligned} \rho'_{\text{fuch}}(z,t) &= \rho_{\text{fu}}(T)\,\theta_{\text{fu}}(z,t) + \rho_{\text{fv}}(T)\,\theta_{\text{vg}}(z,t) \\ &= \frac{\rho_{\text{fu}}(T)\,A_{\text{fu}}(z,t)}{\text{AXMX}} + \frac{\rho_{\text{fv}}(T)\,A_{\text{vg}}(z,t)}{\text{AXMX}} \end{aligned} \tag{14.4-6}$$

where

ρ_{fuch} is the generalized smear density of all the mobile fuel (solid, liquid and vapor) in the channel;

ρ_{fu} is the theoretical density of the liquid or solid fuel;

ρ_{fv} is the physical fuel vapor density;

$$\rho'_{\text{ff}} = \rho_{\text{ff}}(T)\,\theta_{\text{ff}}(z,t) \tag{14.4-7}$$

where

ρ'_{ff} is the generalized smear density of the frozen fuel crust.

$$\rho'_{\text{Na}} = \rho_{\text{N1}}(T)\,\theta_{\text{N1}}(z,t) + \rho_{\text{Nv}}(T)\,\theta_{\text{vg}}(z,t) \tag{14.4-8}$$

where

ρ_{Na} is the generalized smear density of all the sodium (moving liquid, stationary liquid film, and vapor).

$$\rho'_{\text{Nm}} = \rho'_{\text{Na}} - \rho_{\text{N1}}(T)\,\theta_{\text{fm}}(z,t) \tag{14.4-9}$$

where

ρ_{Nm} is the generalized smear density of the sodium vapor and the mobile liquid sodium.

$$\rho'_{\text{fi}} = \rho_{\text{fi}}(T,p)\,\theta_{\text{vg}}(z,t) \tag{14.4-10}$$

where

ρ_{fi} is the generalized smear density of the free fission gas in the channel.

$$\rho'_{\text{fi}} = \rho_{\text{fv}}(T,p)\,\theta_{\text{vg}}(z,t) \tag{14.4-11}$$

where

ρ'_{fv} is the generalized smear density of the fuel vapor in the channel.

$$\rho'_{\text{Mi}} = \rho'_{\text{Nm}} + \rho'_{\text{fi}} + \rho'_{\text{fv}} \tag{14.4-12}$$

where

ρ'_{Mi} is the total generalized smear density of all moving "light components" (liquid sodium, sodium vapor, free fission gas, and fuel vapor).

$$\rho'_{\text{fs}} = \rho_{\text{fs}}\theta_{\text{fu}}(z,t) \tag{14.4-13}$$

where

ρ'_{fs} is the generalized smear density of the dissolved fission gas in the moving liquid or solid fuel.

All the variables and subscripts are described in the list of symbols at the beginning of Chapter 14. The subscript **ch**, **op** requires some additional explanation. It designates the open channel cross section which is occupied by the moving materials in the channel including the sodium film. The basic definition of $\theta_{\text{ch,op}}$ will become clearer if one rewrites Eq. (14.3-3) with $\theta_{\text{ch,op}}$ on the left-hand side of the equation.

In Eq. (14.4-6), the definition of the volume fractions, which is given in Eq. (14.4-1), is used to rewrite the right-hand side of this equation. The initial forms of the conservation equations described in the following subsections contain products of the physical densities and the cross-sectional areas associated with each component. After dividing these equations by AXMX, one can make use of the above definitions for the volume fractions and generalized linear densities to simplify the conservation equations.

For the generalized source or sink terms,

$$S' = \frac{S^l}{\text{AXMX}} \tag{14.4-14}$$

where the source or sink terms S^l are the mass sources and sinks per unit time and per unit length.

14.4.2 Mass Conservation for Fuel, Sodium, and Free and Dissolved Fission-gas

14.4.2.1 Moving Components: Differential Equation

14.4.2.1.1 Fuel and Fuel Vapor Mass Conservation

The differential equation for the mass conservation of all the moving fuel, including fuel vapor*[4] is:

$$\frac{\partial}{\partial \mathrm{t}}\left(\rho_{\mathrm{fu}} A_{\mathrm{fu}}+\rho_{\mathrm{fv}} A_{\mathrm{vg}}\right)+\frac{\partial}{\partial \mathrm{z}}\left(\rho_{\mathrm{fu}} A_{\mathrm{fu}} u_{\mathrm{fu}}+\rho_{\mathrm{fv}} A_{\mathrm{vg}} u_{\mathrm{Mi}}\right) = \left(S^{l}_{\mathrm{fu,ej}}+S^{l}_{\mathrm{fu,me}}-S^{l}_{\mathrm{fu,ff}}\right) \tag{14.4-15}$$

where the source terms denoted by S^l are the mass sources and sinks per unit time and per unit length. The source term due to fuel ejection has been discussed in Section 14.3. The source and sink terms due to fuel freeze-out and frozen fuel remelting, respectively, will be discussed in Section 14.4.3.2.

By dividing the above mass conservation equation by AXMX and by using the definitions for the generalized volume fractions and smear densities one arrives at:

$$\frac{\partial}{\partial \mathrm{t}}\rho'_{\mathrm{fuch}}+\frac{\partial}{\partial \mathrm{z}}\left(\rho'_{\mathrm{fuch}} u_{\mathrm{fu}}\right)+\frac{\partial}{\partial \mathrm{z}}\left(\rho'_{\mathrm{fv}} u_{\mathrm{Mi}}\right)-\frac{\partial}{\partial \mathrm{z}}\left(\rho'_{\mathrm{fv}} u_{\mathrm{fu}}\right) = S'_{\mathrm{fu,ej}}+S'_{\mathrm{fu,me}}-S'_{\mathrm{fu,ff}} \tag{14.4-16}$$

where

$$\rho'_{\mathrm{fuch}}=\rho'_{\mathrm{fu}}+\rho'_{\mathrm{fv}} \tag{14.4-17}$$

is the total generalized mobile fuel smear density in the channel including solid and liquid fuel and fuel vapor. The primed source and sink terms represent the mass sources and sinks per unit time and unit volume. The latter is a m^3 of the cell volume AXMX $\cdot \Delta z$ in which all components (including pin, structure and channel components) are assumed to be uniformly smeared.

Fuel vapor streaming into regions with no fuel or with fuel which is not generating significant fuel vapor pressure (i.e., less than 10^{-2} MPa) is not currently considered in PLUTO2.

14.4.2.1.2 Dissolved Fission-gas Mass Conservation

The mass conservation for the dissolved fission gas (or matrix fission gas) in the moving liquid or solid fuel reads:

$$\frac{\partial}{\partial \mathrm{t}}\left(\rho_{\mathrm{fs}} A_{\mathrm{fu}}\right)+\frac{\partial}{\partial \mathrm{z}}\left(\rho_{\mathrm{fs}} A_{\mathrm{fu}} u_{\mathrm{fu}}\right) = \left[\left(S^{l}_{\mathrm{fu,ej}}\rho_{\mathrm{fsca}} A_{\mathrm{fuca}}\right) \Big/ \left(\rho_{\mathrm{fuca}} A_{\mathrm{fuca}}\right)-\frac{\left(S^{l}_{\mathrm{fu,ff}}\rho_{\mathrm{fs}} A_{\mathrm{fu}}\right)}{\left(\rho_{\mathrm{us}} A_{\mathrm{fu}}\right)}-S^{l}_{\mathrm{fs,rl}}\right] \tag{14.4-18}$$

where the fuel mass sources $S^l_{\mathrm{fu,\,ej}}$ and $S^l_{\mathrm{fu,\,ff}}$ have been multiplied by the dissolved gas-to-fuel mass ratios in the molten pin cavity and in the channel, respectively, to obtain the dissolved gas sources. The areas and densities with the subscripts **fsca** and **fuca** are for the molten pin cavity. The sink term $S^l_{\mathrm{fs,\,rl}}$ is the rate of dissolved

[4] **Fuel vapor is not yet operational.

fission-gas release. Its generalized form is described in Eq. (14.4-20). By dividing Eq. (14.4-18) by AXMX and by using the definitions for the generalized smear densities, one arrives at:

$$\frac{\partial}{\partial t}\rho'_{fs} + \frac{\partial}{\partial z}\rho'_{fs}u_{fu} = \left[\frac{(S'_{fu,ej}\rho'_{fsca})}{\rho'_{fuca}} - \frac{(S'_{fu,ff}\rho'_{fs})}{\rho'_{fu}} - S'_{fs,rl}\right] \tag{14.4-19}$$

$$S'_{fs,rl} = \begin{cases} \rho'_{fs} \cdot \text{CIRTFS} & \text{for } T_{fu} > T_{fu,sol} \\ 0 & \text{for } T_{fu} < T_{fu,sol} \end{cases} \tag{14.4-20}$$

where CIRTFS is a dissolved gas release time constant which is input and has the dimensions s^{-1}. The same gas release time constant is used for releasing the dissolved gas in the molten pin cavity (see Section 14.2.6).

14.4.2.1.3 Two-phase Sodium Mass Conservation

The mass conservation equation of the sodium liquid and vapor including the sodium film is given by

$$\frac{\partial}{\partial t}\left(\rho_{Na}A_{ch}\right) + \frac{\partial}{\partial z}\left(\rho_{Nm,ch}A_{Nm}u_{Mi}\right) = 0 \tag{14.4-21}$$

where

ρ_{Na} is the total sodium smear density, including the sodium films, in the channel. Note that this is not a generalized smear density.

$\rho_{Nm,ch}$ is the smear density of the mobile sodium referring to the cross section A_{Nm} of the moving sodium mixture.

$$A_{Nm} = A_{ch} - A_{Na,fm} - A_{fu} - A_{ff} \tag{14.4-22}$$

where

A_{ch} is the cross sectional area of the channel;

$A_{Na,fm}$ is the cross sectional area of the liquid Na films;

A_{ff} is the cross sectional area of frozen fuel.

By dividing the mass conservation Eq. (14.4-21) by AXMX and using the definitions for the generalized smear densities, one obtains:

$$\frac{\partial}{\partial t}\rho'_{Na} + \frac{\partial}{\partial z}\rho'_{Nm,ch}u_{Mi} = 0 \tag{14.4-23}$$

where

$$\rho'_{Nm} = \rho'_{Na} - \rho_{Nl}\theta_{fm} \tag{14.4-24}$$

The calculation of the volume fraction of the sodium film, θ_{fm}, considers vapor condensation, film evaporation and film entrainment. This is described in Section 14.4.2.2.

14.4.2.1.4 Free Fission-gas Mass Conservation

$$\frac{\partial}{\partial \mathrm{t}}\left(\rho_{\mathrm{fi}} A_{\mathrm{vg}}\right)+\frac{\partial}{\partial \mathrm{z}}\left(\rho_{\mathrm{fi}} A_{\mathrm{vg}} u_{\mathrm{Mi}}\right)=\left(S_{\mathrm{fi,ej}}^{l}+S_{\mathrm{fs,rl}}^{l}\right) \tag{14.4-25}$$

where the two source terms are due to fission-gas ejection from the fuel pins and due to the release of fission gas dissolved in the fuel. The rate of fuel ejection from the pins is described in Section 14.3. The release of the dissolved fission gas is described in Eq. (14.4-20). Dividing by AXMX and using the generalized smear density definitions, one obtains:

$$\frac{\partial}{\partial \mathrm{t}} \rho'_{\mathrm{fi}}+\frac{\partial}{\partial \mathrm{z}} \rho'_{\mathrm{fi}} u_{\mathrm{Mi}}=S'_{\mathrm{fi,ej}}+S'_{\mathrm{fs,rl}} \tag{14.4-26}$$

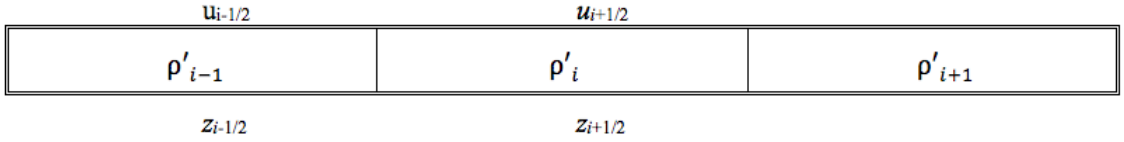
14.4.2.2 Finite Difference Forms of the Mass Conservations and Subroutine PLMACO

For the free fission gas and the dissolved fission gas, the form of the differential mass conservation is:

$$\frac{\partial \rho'_{\mathrm{k}}}{\partial \mathrm{t}}+\frac{\partial\left(\rho'_{\mathrm{k}} u_{\mathrm{k}}\right)}{\partial \mathrm{z}}=\sum_{\mathrm{n}} S_{\mathrm{k,n}} \tag{14.4-27}$$

where k designates a specific component and n the different types of source or sink terms for this component.

For the numerical solution of these equations, a staggered mesh with the velocities at the edges and the densities at the mesh centers is used (on the numerical grid, which is actually used in the code, the half indices have to be avoided; see later in this section).



By spatially integrating over the control volume for $z_{\mathrm{i\text{-}1/2}}$ to $z_{\mathrm{i+1/2}}$ one obtains

$$\frac{\partial \rho'_{\mathrm{k,i}}}{\partial \mathrm{t}} \Delta z_{\mathrm{i}}+\left[\left(\rho' u\right)_{\mathrm{k,i+1/2}}-\left(\rho' u\right)_{\mathrm{k,i\text{-}1/2}}\right]=\sum_{\mathrm{n}} S'_{\mathrm{k,n,i}} \Delta z_{\mathrm{i}} \tag{14.4-28}$$

By performing the time integration over the length of the time step, Δt, we obtain:

$$\rho'^{n+1}_{\mathrm{k,i}}=\rho'^{n}_{\mathrm{k,j}}-\left[\left(\rho' u\right)_{\mathrm{k,i+1/2}}-\left(\rho' u\right)_{\mathrm{k,i\text{-}1/2}}\right] \frac{\Delta t}{\Delta z_{\mathrm{i}}}+\sum_{\mathrm{n}} S'_{\mathrm{k,n,i}} \cdot \Delta t \tag{14.4-29}$$

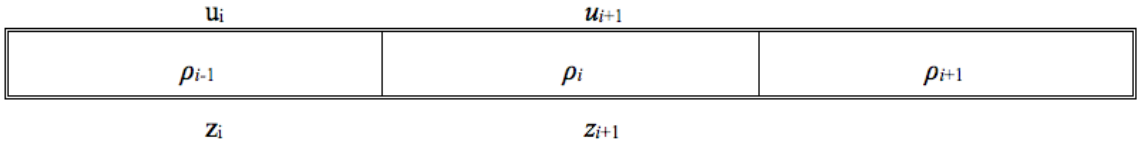
It should be noted that this finite difference form of mass conservation is in a conservation form that prevents any mass losses. It should be noted that this mass conservation actually includes a variable flow cross-section treatment because the primed generalized smear densities include the flow cross-section areas.

Full donor cell differencing is used for evaluating the convective fluxes. Although this is not as accurate as higher order approaches, it tends to increase the stability of the solution [14-30, 14-31]. This is important because of the very complicated flow condition which can be encountered in PLUTO2 calculations. The convective terms in Eq. (14.4-29) are calculated in the following manner:

$$\left(\rho' u\right)_{\mathrm{k,i+1/2}}=\begin{cases}\rho'_{\mathrm{k,i}} u_{\mathrm{k,i+1/2}} & \text{if } u_{\mathrm{k,i+1/2}}>0 \\ \rho'_{\mathrm{k,i+1}} u_{\mathrm{k,i+1/2}} & \text{if } u_{\mathrm{k,i+1/2}}<0\end{cases} \tag{14.4-30}$$

$$(\rho' u)_{\text{k,i-1/2}} = \begin{cases} \rho'_{\text{k,i}} u_{\text{k,i-1/2}} & \text{if } u_{\text{k,i-1/2}} > 0 \\ \rho'_{\text{k,i+1}} u_{\text{k,i-1/2}} & \text{if } u_{\text{k,i-1/2}} < 0 \end{cases} \tag{14.4-31}$$

In the above equations, half indices are used that can be located on the previous schematic of the mesh grid. However, since half indices cannot be used in a computer program, the indices used in the code are arranged on the numerical grid as follows:



When using FORTRAN variable names, the convective flux for the free fission gas at $i - 1/2$ is therefore:

$$\text{COFICH}\,(I) = \begin{cases} \text{DEFICH}\,(I-1) \cdot \text{UMCH}\,(I) & \text{if UMCH}\,(I) > 0 \\ \text{DEFICH}\,(I) \cdot \text{UMCH}\,(I) & \text{if UMCH}\,(I) < 0 \end{cases} \tag{14.4-32}$$

The convective fluxes are set to zero at the edges of the component domains. The location of the upper and lower interfaces of each component domain is calculated in PLUTO2. The integers designating the lowermost and uppermost nodes containing free fission gas are designated IFFIBT and IFFITP, respectively, and the convective fluxes are set to zero at these locations:

$$\text{COFICH}\,(\text{IFFIBT}) = 0 \tag{14.4-33}$$

$$\text{COFICH}\,(\text{IFFITP} + 1) = 0 \tag{14.4-34}$$

Eq. (14.4-15) is the fuel mass conservation in differential form. It can be rearranged to give

$$\begin{aligned} \frac{\partial}{\partial \text{t}} \rho'_{\text{fuch}} + \frac{\partial}{\partial \text{z}} \left[\left(\rho'_{\text{fuch}} - \rho'_{\text{fv}} \right) u_{\text{fu}} \right] + \frac{\partial}{\partial \text{z}} \left(\rho'_{\text{fv}} u_{\text{Mi}} \right) \\ = S'_{\text{fu,ej}} + S'_{\text{fu,me}} - S'_{\text{fu,ff}} \end{aligned} \tag{14.4-35}$$

By integrating over Δz_{i} and Δt, one obtains:

$$\begin{aligned} \rho'^{n+1}_{\text{fuch,i}} = \rho'^{n}_{\text{fuch,j}} - \Big\{ \left[\left(\rho'_{\text{fuch}} - \rho'_{\text{fv}} \right) u_{\text{fu}} \right]_{\text{i+1/2}} - \left[\left(\rho'_{\text{fuch}} - \rho'_{\text{fv}} \right) u_{\text{fu}} \right]_{\text{i-1/2}} \\ + \left(\rho'_{\text{fv}} u_{\text{Mi}} \right)_{\text{i+1/2}} - \left(\rho'_{\text{fv}} u_{\text{Mi}} \right)_{\text{i-1/2}} \Big\} \frac{\Delta t}{\Delta z_{\text{i}}} \\ + \left[S'_{\text{fu,ej,i}} + S'_{\text{fu,me,i}} - S'_{\text{fu,ff,i}} \right] \Delta t \end{aligned} \tag{14.4-36}$$

In the code, the convective flux term $\left[\left(\rho'_{\text{fuch}} - \rho'_{\text{fv}} \right) u_{\text{fu}} \right]_{\text{i-1/2}}$ is written as:

$$\text{COFUCH}\,(I) = \begin{cases} \left[\text{DEFICH}\,(I-1) - \text{DEFVCH}\,(I-1) \right] \cdot \text{UFCH}\,(I) & \text{if UFCH}\,(I) > 0 \\ \left[\text{DEFUCH}\,(I) - \text{DEFVCH}\,(I) \right] \cdot \text{UFCH}\,(I) & \text{if UFCH}\,(I) < 0 \end{cases} \tag{14.4-37}$$

The two-phase sodium mass conservation equation, Eq. (14.4-23), includes the generalized smear density of the moving mixture in the convective term and not the total sodium smear density which includes the sodium film. The value of the density is evaluated according to Eq. (14.4-24). When using the FORTRAN variable names, the convective flux at the boundary $i - 1/2$ is:

$$\text{CONACH}\,(I) = \begin{cases} \left[\text{DEFICH}\,(I-1) \right] \cdot \text{UMCH}\,(I) & \text{if UMCH}\,(I) > 0 \\ \left[\text{DENMCH}\,(I) \right] \cdot \text{UMCH}\,(I) & \text{if UMCH}\,(I) < 0 \end{cases} \tag{14.4-38}$$

All of the above mass conservation equations are solved in subroutine PLMACO (**P**LUTO2 **M**ASS **C**ONSERVATION). However, sink or source terms related to fuel plateout and frozen fuel crust release are

considered earlier in the calculational sequence in subroutine PLFREZ. Fuel and fission-gas ejection terms are considered later when subroutine PL1PIN, which calculates these terms, is called. Besides solving the mass conservation equations, PLMACO also contains a special treatment for the numerical cells at the bottom and top of the channel, which comes into play only when the interaction zone extends to the subassembly inlet or outlet. For example if the interaction zone has reached the subassembly outlet and if the pressure in the top node is larger than the outlet pressure, appropriate amounts of fuel, sodium and fission gas are taken out of the top nodes in order to reduce the pressure to the outlet pressure. The same procedure is performed for the inlet node if the interaction zone extends into it. The total component masses taken out of the end nodes are shown in the PLUTO2 output.

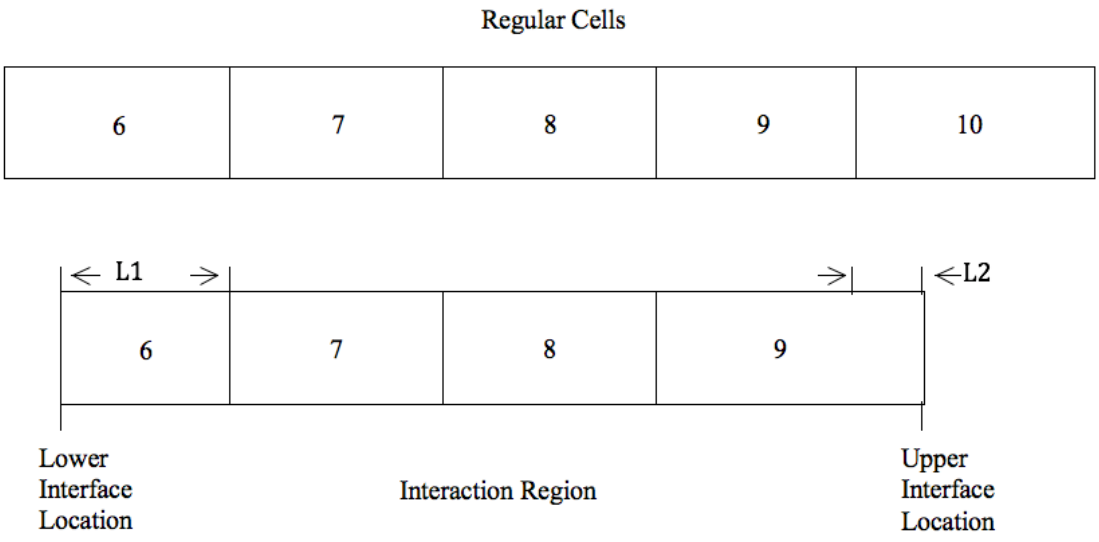
14.4.2.3 Determination of the Axial Extent of the Component Regions in Subroutine PLIF and Mesh Rezoning in Subroutine PLREZO

Before the above mass conservation equations are solved in subroutine PLMACO, the axial extent of the different component regions has to be determined. In subroutine PLIF (**PL**UTO2 **I**NTERFACE), the interface locations of the interaction region are reset (the velocity calculations of the liquid sodium slugs above and below this region, which determine the interface velocities, are actually done later in the sequence in PLMOCO). The resetting of the interaction zone interface locations is not quite straightforward because the sodium slugs leave a liquid sodium film behind on the cladding and structure when the interaction zone expands. Because the dynamic slug calculation in PLMOCO is done for the entire channel cross section area, the interface displacement is increased by a factor of 1/(1 - liquid film cross section/channel cross section). This conserves the sodium voiding calculated for the entire cross section. For a sodium slug reentering over an existing film the slug interface displacement is increased by multiplying by (1 + liquid film cross section/channel cross section). PLIF also resets the lower and upper interface locations of the fuel region which can be inside the interaction region or at the interaction zone boundaries. If the latter are being penetrated by the fuel, they are set to the fuel interfaces in subroutine PLREZO, as discussed below (the actual velocity calculation of the fuel interfaces is done in PLMOCO). Furthermore, the upper and lower interface locations of both the free fission gas and the fuel vapor region are calculated in the subroutine PLIF (these interfaces can also be at the edges of the interaction region or inside it; the velocities of these interfaces are assumed to be equal to the local sodium/fission gas/fuel vapor mixture velocities). Besides calculating the extent of the component regions, the subroutine PLIF also calculates the axial fuel-pin failure propagation (see Section 14.3). If the fuel pins fail into the liquid sodium slugs outside the interaction zone, the interaction zone is enlarged in subroutine PLREZO. This is discussed below.

In subroutine PLREZO (**PL**UTO2 **REZ**ONE), mesh cells are added to an expanding interaction region or deleted from a shrinking interaction region. In the schematic below it is shown that the end cells of the interaction region can be shorter or longer than the regular cell lengths. The small cell 6 in the schematic exists because L1 is greater than the input value DZPLIN, which has to be shorter than any regular mesh cell used in a given calculation. Cell 9 is larger than the regular cell length at this location because it has not yet extended enough into cell 10 (L2 < DZPLIN). Creating a new cell or collapsing a small cell with a neighboring one is a complicated procedure because it requires redistributing liquid sodium and fission gas in a manner that no significant pressure disturbances are introduced. The SAS4A reactivity calculation refers only to the regular mesh grid and not the Lagrangian edge cells. The material in a particular cell grid such as cell No. 6 is assumed to be homogeneously mixed with the liquid sodium sluglet to the left of it for the reactivity calculation. The materials in the section L2 of cell 9 are assumed to be homogeneously mixed with cell 10 for the reactivity calculation.

Subroutine PLREZO also expands the interaction zone when fuel penetrates the liquid sodium slugs. The liquid sodium that has been penetrated by the fuel is added to the edge cell and homogeneously distributed into it.

PLREZO also has the following treatment for slug interfaces (with the interaction region) crossing into cells that contain frozen fuel crusts or cells into which fuel can be ejected from adjacent ruptured fuel pin nodes: the liquid sodium that has crossed into such a cell is homogeneously distributed in it and the slug interface location is reset to the edge of the cell containing frozen fuel or the ejection cell, respectively.



14.4.2.4 Stationary Sodium Films and Subroutine PLVOFR

The liquid sodium films that are left behind by the ejected coolant slugs are of importance for keeping the cladding cool in voided regions of the coolant channel. Moreover, when fuel moves into voided regions in which a liquid sodium film is still present, the fuel-to-liquid sodium heat transfer will increase the liquid sodium evaporation, and thus the sodium vapor pressure.

In PLUTO2, the liquid sodium film is assumed to be stationary. It can become thinner due to sodium evaporation or to entrainment caused by the drag from the moving vapor/gas/sodium droplet/fuel particle mixture. Once an annular fuel flow regime develops in a coolant channel node, the entire sodium film in that node is added to the moving vapor/gas/sodium droplet flow (This assumption should be further investigated). Liquid sodium films can grow due to sodium vapor condensation and sodium droplet de-entrainment, i.e., deposition on the film. However, the maximum thickness of the liquid film cannot be larger than the initial film thickness, which is determined by the input liquid film fraction CINAFO.

The evaporation of, or the condensation on, the liquid sodium film is taken into account in the sodium energy equation and contributes to the change of the liquid sodium fraction in a numerical cell. Once the liquid sodium fraction is below the input film fraction, the sodium film fraction will approach and eventually be equal to the current liquid fraction if the node considered is in the de-entrainment mode. If the sodium film is being entrained, the sodium film thickness is determined by this mechanism.

In order to determine whether sodium entrainment or de-entrainment occurs, the superficial velocity (i.e., the volumetric flux divided by the entire channel cross section) of the fission-gas/two-phase sodium/fuel particle mixture has to be evaluated. The following momentum averaging suggested by Ishii [14-58] is performed to arrive at an average velocity of all the moving components:

$$\overline{u}_{\mathrm{Mi,fu}} = \left\{ \frac{\left| \left(\mathrm{CIETFU} \cdot \frac{\rho'_{\mathrm{fu}}}{\theta_{\mathrm{ch}}} \cdot u_{\mathrm{fu}} \left| u_{\mathrm{fu}} \right| + \frac{\rho'_{\mathrm{Mi}}}{\theta_{\mathrm{ch}}} \cdot u_{\mathrm{Mi}} \left| u_{\mathrm{Mi}} \right| \right) \right|}{\left(\mathrm{CIETFU} \cdot \frac{\rho'_{\mathrm{fu}}}{\theta_{\mathrm{ch}}} + \frac{\rho'_{\mathrm{Mi}}}{\theta_{\mathrm{ch}}} \right)} \right\}^{1/2} \tag{14.4-39}$$

where CIETFU is an input value which can be between zero and one. This input variable has been introduced because of the uncertainty about the influence of the fuel particles on the entrainment of the film. If one assumes that the fuel particle mass can be smeared over the entraining mixture cross section and that it would act like a dense gas, CIETFU $= 1.0$ has to be chosen. If one assumes that the fuel particles are less efficient in entraining

the film, a lower value should be used. A value of 0.1 is recommended because it was used successfully in experiment analyses of TREAT tests L8 and H6 [14-15, 14-12].

The entrainment limit $u_{\text{et,min}}$ used in the code is based on a correlation given in Reference [14-33, 14-34] for the inception of entrainment of a rough turbulent film flow:

$$u_{\text{et,min}} = \begin{cases} \text{VCONST}^{0.8} \cdot \frac{\sigma_{\text{N1}}}{\mu_{\text{N1}}} \cdot \sqrt{\frac{\rho_{\text{N1}}}{\rho_{mi,fu}}} & \text{if VCONST} < 1/15 \\ 0.1146 \cdot \frac{\sigma_{\text{N1}}}{\mu_{\text{N1}}} \cdot \sqrt{\frac{\rho_{\text{N1}}}{\rho_{\text{mi,fu}}}} & \text{if VCONST} > 1/15 \end{cases} \tag{14.4-40}$$

where the viscosity number VCONST is defined by

$$\text{VCONST} = \frac{\mu_{\text{N1}}}{\left\{ \rho_{\text{N1}} \sigma_{\text{N1}} \left[\frac{\sigma_{\text{N1}}}{(g \cdot (\rho_{\text{N1}} - \rho_{\text{Mi,fu}}))} \right]^{1/2} \right\}^{1/2}} \tag{14.4-41}$$

μ_{Nl} = viscosity of liquid sodium

σ_{N1} = surface tension of liquid sodium

ρ_{N1} = density of liquid sodium

$\rho_{\text{Mi,fu}} = \text{CIETFU} \cdot \frac{\rho'_{\text{fu}}}{\theta_{\text{ch}}} + \frac{\rho'_{\text{Mi}}}{\theta_{\text{ch}}}$

In the entrainment mode:

$$|\overline{u}_{\text{Mi,fu}}| > u_{\text{et,min}} \tag{14.4-42}$$

The amount of sodium film entrained is computed using the following equation.

$$\theta^{\text{new}}_{\text{Mi,fu}} = \theta^{\text{old}}_{\text{fm}} - \frac{\text{CINAFO} \cdot \theta_{\text{ch,op}} \cdot \Delta t_{\text{PL}}}{5 \text{ ms}} \tag{14.4-43}$$

where

CINAFO $\cdot \theta_{\text{ch,op}}$ = initial and maximum liquid sodium film generalized volume fraction (CINAFO is input);

$\Delta t_{\text{PL}} / (5 \text{ ms})$ = fraction of the initial and maximum film volume fraction which is entrained during a PLUTO2 time step.

The entrainment time constant of 5 milliseconds will lead to a complete entrainment of the liquid film in 5 milliseconds if the mixture velocity stays above the entrainment limit during that time. Since the entrained liquid sodium droplets are assumed to be instantaneously in velocity equilibrium with the moving mixture, the velocity of the latter can decrease due to the entrainment. This may temporarily lead to a de-entrainment period, which is characterized by:

$$\overline{u}_{\text{Mi,fu}} < u_{\text{et, min}}$$

and

$$\theta^{\text{new}}_{\text{fm}} = \theta^{\text{old}}_{\text{fm}} + \frac{\text{CINAFO} \cdot \theta_{\text{ch,op}} \cdot \Delta t_{\text{PL}}}{5 \text{ ms}} \tag{14.4-44}$$

Subroutine PLVOFR. Subroutine PLVOFR (**PL**UTO2 **VO**LUME **FR**ACTIONS) calculates the sodium entrainment/de-entrainment discussed above. Two other tasks are performed in this routine. One is concerned with the setting of most channel volume fractions and the other with the selection of the fuel flow regimes. The latter will be discussed in the next section.

Since PLVOFR is called after PLMACO and PLFREZ, the volume fractions calculated in PLVOFR already reflect the component mass changes due to convection and due to fuel plateout and fuel crust release. Volume

fractions set in this routine include the moving fuel volume fraction θ_{fu}, the volume fraction of the open channel $\theta_{\text{ch,op}} = \theta_{\text{ch}} - \theta_{\text{ff}}$, the liquid sodium volume fraction θ_{Nl}, the vapor/gas volume fraction when the liquid sodium is assumed to be uncompressed $\theta_{\text{vg,un}}$, the vapor/gas volume fraction for properly compressed liquid sodium θ_{vg}, and also the sodium film volume fraction θ_{fm} as discussed above. Moreover, the sodium quality is calculated in this routine. It is based on the following definition:

$$\frac{1}{\rho_{\text{Na}}} = \frac{x_{\text{Na}}}{\rho_{\text{Nv}}} + \frac{(1 - x_{\text{Na}})}{\rho_{\text{N1}}} \tag{14.4-45}$$

The last equation can be solved for x_{Na}:

$$x_{\text{Na}} = \frac{(\rho_{\text{N1}} - \rho_{\text{Na}}) \cdot \rho_{\text{Nv}}}{(\rho_{\text{N1}} - \rho_{\text{Nv}}) \cdot \rho_{\text{Na}}} \tag{14.4-46}$$

where

$$\rho_{\text{Na}} = \frac{\rho'_{\text{Na}}}{(\theta_{\text{ch,op}} - \theta_{\text{fu}})}$$

The sodium void fraction is also set in PLVOFR. It is based on the definition

$$\rho_{\text{Na}} = \alpha_{\text{Na}} \rho_{\text{Nv}} + (1 - \alpha_{\text{Na}})\, \rho_{\text{N1}} \tag{14.4-47}$$

This leads to

$$\alpha_{\text{Na}} = \frac{(\rho_{\text{N1}} - \rho_{\text{Na}})}{(\rho_{\text{N1}} - \rho_{\text{Nv}})} \tag{14.4-48}$$

The volume fraction of the vapor/gas mixture for the case of uncompressed liquid sodium is

$$\theta_{\text{vg,un}} = (\theta_{\text{ch,op}} - \theta_{\text{fu}}) \cdot \alpha_{\text{Na}} \tag{14.4-49}$$

The gas/vapor volume fraction for compressed liquid sodium is

$$\theta_{\text{vg}} = \theta_{\text{vg,un}} + K_{\text{N1}} \cdot \theta_{\text{N1}} \cdot (P_{\text{ch}} - P_{\text{Nv}}) \tag{14.4-50}$$

where

$K_{\text{N1}} =$ CMNL which is a single input value for the isothermal liquid sodium compressibility.

The θ_{vg} calculated in this routine is not allowed to be smaller than $\frac{\theta_{\text{ch,op}}}{1000}$. This is done because θ_{vg} is used in the denominator of certain interaction terms. Later in the fission-gas pressure calculation in subroutine PLNAEN, the θ_{vg} is implicitly used without the above-mentioned restriction.

14.4.3 Fuel Flow Regimes, Fuel Plateout and Frozen Crust Release, Plated-out and Moving Fuel Configurations, and Energy and Momentum Exchange Terms

14.4.3.1 Fuel Flow Regimes

Based on the evidence from TREAT TOP experiments, only a fraction of the fuel ejected into the coolant channels fragments into particles or droplets that get rapidly swept upwards. The other fuel does not break up but moves in the channels in the form of a continuous flow that tends to plate out on the cladding and structure upon freezing. Particulate debris with only little plated-out fuel was found in TOP tests R12 [14-35] and J1 [14-25] in which the power transients were cut soon after the first indication of pin failure. This suggests that

the fuel that is ejected first from the pins and contacts liquid sodium tends to break up. The fuel that is ejected later into a locally voided channel tends to stay together although liquid sodium reentering from below can cause a delayed fragmentation. This was concluded from the analysis of TREAT test H4 [14-37, 14-36].

The CAMEL out-of-pile tests [14-38, 14-39] also show rapid particulate fuel sweepout. The fuel that does not fragment tends to plate out very close to the fuel ejection site in these out-of-pile tests. The regions with plated-out fuel found in the post-test examination of most TREAT TOP tests, however, extend considerably above the cladding failure site. This indicates the temporary existence of a continuous fuel flow in TREAT experiments. The likely reasons why the plated-out fuel does not extend further upwards in CAMEL tests are (a) the lack of continued fission heating of the fuel in these out-of-pile tests, (b) the amount of fuel ejected into the channels in these tests was relatively small, and thus, had little stored heat, and (c) the relatively cold cladding and structure in these tests may rapidly cool the fuel. Two further items may have promoted the rather localized plateout in the CAMEL tests. First, the ejection of the thermite fuel in several CAMEL tests was accompanied by excessive amounts of gas. Second, the CAMEL ejection pressure was high for longer times than expected for fuel ejection from prototypic pins. Both these items cause coolant channel voiding and lack of fragmentation.

In PLUTO2, one particulate and two continuous fuel flow regimes are modeled. The latter include annular flow (which can be a total or a partial annular flow) and bubbly flow. Flow regime in a cell is determined based on flow conditions in that cell, and can vary from cell to cell. All possible PLUTO2 flow regimes are depicted in Figure 14.4.1. The use of these flow regimes has the advantage of providing a geometry which allows a more straighforward determination of the mass, momentum, and energy exchange terms than in models not explicitly treating different flow regimes.

The particulate or droplet flow regime has been traditionally assumed in fuel-coolant interaction (FCI) models such as the Cho-Wright model [14-9], the Board-Hall model [14-40], and the MURTI model [14-41]. Whole-core modules assuming particulate flow include the SAS/FCI module of SAS3D [14-1] and the EPIC code [14-7, 14-8] which has been coupled with the SAS3D code. It is also interesting to note that the SIMMER-II disassembly and transition phase code [14-29] treats all its moving liquid or solid components as droplet or particulate flows. However, some attempts are made to account for the effect of flow regimes on exchange terms.

The PLUTO code [14-3, 14-4], which is the predecessor of PLUTO2, assumed exclusively particulate or droplet flow. It was nevertheless successful in analyzing the fuel motion and sodium voiding during the first few tens of milliseconds after pin failure in the E8 \$3/s TOP experiment [14-5, 14-6] and also the fuel motion and voiding during the first event of the H6 \$0.5/s TOP experiment [14-6].

The particulate flow regime has therefore been retained in PLUTO2. The fuel ejected into liquid sodium is assumed to instantaneously fragment into droplets or particles of radius RAFPLA if the liquid sodium fraction $\theta_{\mathrm{N1}} > \mathrm{VFNALQ} \cdot \theta_{\mathrm{ch,\,op}}$. Both RAFPLA and VFNALQ are input. All particles can, later on, simultaneously further fragment (after an input time delay of TIFP after pin failure) into smaller particles with radius RAFPSM, which is also input. Moreover, if the liquid sodium fraction in a certain channel node exceeds the input value VFNARE, mobile fuel, which was previously in a continuous flow regime (i.e., annular or bubbly flow), will instantaneously fragment into droplets in this node. This is an attempt to simulate delayed FCIs generated by sodium slugs reentering from below.

The continuous fuel flow regime in PLUT02 will develop if considerable local voiding has occurred ($\theta_{\mathrm{N1}} < \mathrm{VFNALQ} \cdot \theta_{\mathrm{ch,\,op}}$) and if the fuel energy is above the input fuel energy EGMN. For the latter a value somewhat above the solidus energy is suggested. This is because particles with an average energy corresponding to the solidus may already have a frozen outer crust which will prevent them from splattering on cladding and will thus not lead to an annular film flow. For the input value VFNALQ, which is the liquid sodium fraction below which a continuous fuel flow can develop, a value of 0.33 is recommended because it was used in the successful L8 post-test analysis [14-15, 14-12].

Once a continuous fuel flow has developed and is not yet occupying most of the coolant channel open cross sectional area at a certain elevation, a partially or fully annular fuel flow regime is assumed at that elevation. The partially annular flow regime was introduced because it appears unlikely that a relatively small amount of liquid

fuel would spread around the entire perimeter of the coolant channel (see Eq. (14.4-78)). Once the fuel volume fraction at a certain axial elevation becomes higher than the input value CIBBIN (for which a value of 0.7 is recommended), the development of a bubbly fuel flow is assumed to occur. This bubbly flow is then assumed to exist until the fuel volume fraction drops below 2/3 of the input volume fraction CIBBIN. This is because surface tension effects can maintain the bubbly flow down to a lower fuel volume fraction compared to the value required to cause the onset of the bubbly flow. The decision about flow regime changes is made at the end of subroutine PLVOFR. The logic flow for deciding whether the flow regime in a certain node should remain the same or whether it should change to another one is shown in Figure 14.4.4. The sequence of "if" checks and statements is exactly the same as in the program. The input parameters appearing in the flowchart have already been described above. The meaning of the Flow Regime [F.R.] numbers is the following:

F.R. 1 particulate or droplet flow regime

F.R. 3 partially or fully annular fuel flow

F.R. 4 bubbly fuel flow

The flow regime labels for each node i are stored in the code in pointer array IFLAG(I). The value 2 has been reserved for a possible future PLUTO2 version in which cladding motion is considered and in which the number 2 would designate nodes with moving cladding. In the LEVITATE module, flow regime 2 is operational and designates molten cladding flow with imbedded fuel drops or chunks.

In explaining the flow chart logic, it is best to start at the time of PLUTO2 initiation when the flow regime in all nodes is set to 1 (particulate flow). The fuel will remain in flow regime 1 at least until the fourth diamond from above leads to a "no". However, if the fuel particles are already cold enough, the third diamond from above will keep them in particulate form. If both the third and fourth diamond lead to "no", the logic flow will drop straight through, lead to a complete entrainment of the liquid sodium film in the node considered and then switch to the annular flow regime (F.R. = 3). If the lowermost diamond leads to a "no", a switch to the bubbly flow regime (F.R. = 4) will be made. If such a switch occurs, the vapor/gas/sodium droplets mixture velocity will be set to the fuel velocity. This is done at the moment of bubbly flow regime initiation only. This helps to initialize a well-behaved two-fluid bubbly flow calculation. Starting with the previous vapor/gas/sodium droplets mixture velocity from the annular flow, which can be fairly high, can cause problems with the inertial terms in the bubbly flow calculation.

Once an annular or bubbly flow regime has been established in a node, the only way to get back to the particulate or droplet regime is via the first diamond. The latter will make it a particulate flow only if enough liquid sodium has reentered into the node considered. Once a bubbly flow regime has been established, it will get back to the annular flow regime only if the second lowest diamond leads to a "no". Overall, it has been attempted to keep the numerical nodes in the same flow regime for a reasonable length of time. This has helped to stabilize the overall calculation.

14.4.3.2 Fuel Plateout and Frozen Fuel Crust Release in Subroutine PLFREZ

In subroutine PLFREZ (**PL**UTO2 **FRE**EZING ROUTINE), the amount of fuel plating out on cladding and structure as well as the amount of frozen fuel crust released from an underlying melting cladding are calculated. This information is later used in subroutine PLMISC to update the frozen fuel geometry. PLFREZ is called before the mass conservation equations and it can therefore provide updated densities and velocities of the moving components for the mass conservation. (The velocity change of the moving mixture due to frozen crust release is calculated in PLFREZ). However, since PLFREZ uses data from the end of the last time step, it could also have been called at the end of the calculational sequence as it is done in LEVITATE.

Fuel plateout on the cladding and structure of channel node i can only occur if there is either an annular or bubbly fuel flow regime in that node. Solid particles or particles with a reasonably thick solid outer crust are not believed to be able to stick to a cold surface, and molten droplets can exist in PLUTO2 only if there is a significant liquid sodium volume fraction ($\theta_{N1} > \text{VFNALQ} \cdot \theta_{ch,\,op}$; see previous section). The liquid sodium and, in particular,

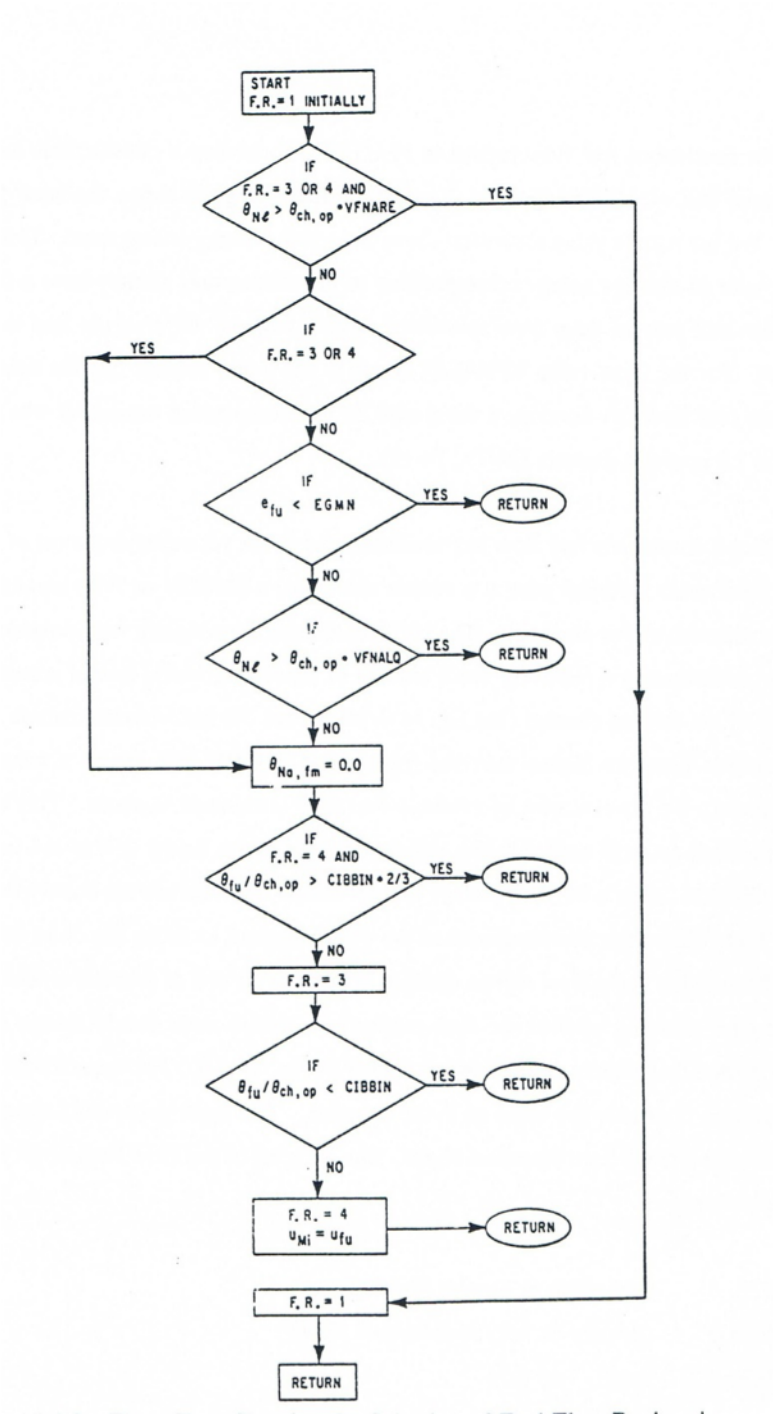


Figure 14.4.4: Flow Chart Showing the Selection of Fuel Flow Regime in a Coolant Channel Mesh Cell

the liquid sodium film are assumed to prevent molten fuel droplets from sticking to cold surfaces because the droplets should form thin solid crusts due to the contact with the liquid sodium. The assumption that liquid droplets which are surrounded by significant amounts of liquid sodium will not plate out on cold surfaces may be questionable when a jet of fuel droplets hits a cold wall with a reasonably high velocity. In this case, the droplets may splatter on the cold wall and form a moving liquid film, which soon freezes and sticks despite the presence of liquid sodium. Such a high-velocity impingement of fuel may have occurred in the P4 test [14-42] or during the later stages of some of the CAMEL tests [14-38, 14-39]. This high-velocity impingement is not modeled in PLUTO2, but its effect can be roughly simulated by using for the input value VFNALQ, which is the sodium liquid fraction above which droplet or particle flow is allowed, a value close to 1, and for the input value VFNARE, also a value close to 1 and greater than VFNALQ. In the P4 pre-test analysis with PLUTO2, VFNALQ = 0.33 was used (similar values had been used in the successful PLUTO2 analyses of H6 [14-6] and L8 [14-15, 14-12]). This input choice led to the prediction of total fuel sweepout because only flow regime 1 (particulate or droplet flow) was selected by the code. The P4 experiment, however, showed extensive plateout and little sweepout. The above explanation of a high-velocity fuel jet hitting the cladding, however, is not the only possible explanation. Another explanation for this preferential plateout may be that the sodium could have bypassed the ejected fuel in this relatively large bundle test and, therefore, not exerted the full inlet pressure on it.

To be in the annular or bubbly flow regime is only one of the necessary conditions for initiating plateout. The other ones are:

$$e_{\text{vu}} < \text{EGBBLY} \tag{14.4-51}$$

and

$$T_{\text{ffcl}} < T_{\text{fu,sol}} \tag{14.4-52}$$

and

$$T_{\text{cl,os}} < \text{TECLMN} \tag{14.4-53}$$

and

$$\theta_{\text{ch,op}} > 0.3\theta_{\text{ch}} \tag{14.4-54}$$

Moreover, the moving fuel volume fraction is not allowed to drop below 0.01 of the channel volume fraction due to fuel plateout. This is done to avoid problems with zero amounts of moving fuel.

The condition 14.4-49 states that the moving that the moving fuel energy should be below the input value EGBBLY which has to be within the melting band of the fuel. A value in the lower part of the melting band is recommended, based on the L8 analysis with PLUTO2 [14-15, 14-12]. One should be careful that the input value EGMN, below which a continuous flow regime cannot develop (see Figure 14.4.4), is smaller than EGBBLY because this can inhibit plateout.

The condition 14.4-50 states that the temperature of the fuel crust on the cladding should be below the fuel solidus temperature in order to allow plateout. Fuel should not plate out on already molten fuel crusts (layers), and, as soon as the fuel crust temperature has increased half way into the melting band, this fuel crust will actually be ablated by the moving fuel field (see below).

The condition 14.4-51 states that fuel plateout will occur only if the cladding surface temperature is less than the input value TECLMN. This value should be set to the steel solidus temperature TESOL (which is input) or somewhat above it. It is believed that freezing fuel will not stick to a molten cladding surface but rather slide over it. However, it will probably also ablate some of the molten steel which is not treated in PLUTO2. If one wants to consider this phenomenon, one will have to switch to the LEVITATE module, which is treating steel ablation. This can be done by setting the input value NCPLEV to a low integer number.

The condition 14.4-51a limits the frozen fuel fraction in a node to 70% of the nodal volume. Thus a complete plugging of a coolant channel is not allowed in PLUTO2. This is done because the PLUTO2 equations are written for relatively smoothly varying flow cross sections. If all the above-mentioned conditions are met in a channel cell, then fuel plateout will occur in that cell. There are two possibilities:

(a) If

$$e_{\text{fu,sol}} < e_{\text{fu}} < \text{EGBBLY} \ ,$$

the amount of mobile fuel which will be plated out in node i during Δt_{PL} is calculated from:

$$\Delta\rho'_{\text{fu,i}} = \rho'_{\text{fu,i}} \cdot \frac{\text{EGBBLY} - e_{\text{fu,i}}}{\text{EGBBLY} - e_{\text{fu,sol}}} \tag{14.4-55}$$

The input parameter (EGBBLY) appeared already in Eq. (14.4-51). It has to be between the fuel solidus and liquidus energies. The latter are the input values EGFUSO and EGFULQ, respectively. It was mentioned earlier that a value in the lower part of the melting range is recommended based on the PLUTO2 L8 analysis [14-15, 14-12]. The fuel which plates out is assumed to have a temperature of

$$T_{\text{fu,fz,i}} = T_{\text{fu,sol}} \tag{14.4-56}$$

This assumption leads to a temperature increase of the mobile fuel when plateout occurs and is accounted for in the code. The underlying assumption is that the mobile fuel has a radial temperature gradient with the freezing fuel being at the fuel solidus temperature.

(b) If

$$e_{\text{fu}} < e_{\text{fu,sol}} \ ,$$

the fraction of the mobile fuel plated out in a PLUTO2 time step of Δt_{PL} second will be calculated from

$$\Delta\rho'_{\text{fu,i}} = \rho'_{\text{fu,i}} \Delta t_{\text{PL}} \cdot \frac{1}{2 \text{ ms}} \tag{14.4-57}$$

which causes a near complete fuel plateout of the mobile fuel in 2 ms. This situation can occur when frozen fuel crusts, which were released in another cell due to melting cladding, slide into a cell in which cladding has not yet melted. The assumption is that these fuel crusts will bring with them some liquid steel that will make them stick on still-unmelted cladding. The temperature of the plated-out fuel in this case equals the temperature of the moving fuel.

$$T_{\text{fu,fz,i}} = T_{\text{fu,i}} \tag{14.4-58}$$

For both cases, the temperature changes of the frozen fuel crusts on cladding and structure are calculated in the same way by making use of the assumptions about $T_{\text{fu, fz}}$ in Eq. (14.4-56) and Eq. (14.4-58).

$$T^{\text{new}}_{\text{ffci,i}} = \frac{T_{\text{fu,fz,i}} \Delta\rho'_{\text{fu,i}} + T^{\text{old}}_{\text{ffcl,i}} \rho'_{\text{ff,i}}}{\rho'_{\text{ff,i}} + \Delta\rho'_{\text{fu,i}}} \tag{14.4-59}$$

and

$$T^{\text{new}}_{\text{ffsr,i}} = \frac{T_{\text{fu,fz,i}} \Delta\rho'_{\text{fu,i}} + T^{\text{old}}_{\text{ffsr}} \rho'_{\text{ff,i}}}{\rho'_{\text{ff,i}} + \Delta\rho'_{\text{fu,i}}} \tag{14.4-60}$$

Because the relative fractions of the fuel plating out on cladding and on structure and the relative fractions of fuel already plated out on cladding and structure are assumed to be the same, the above two equations are very similar. This is because the fraction of the fuel plating out on cladding cancels out in Eq. (14.4-59), and the fraction of fuel plating out on the structure cancels out in Eq. (14.4-60). Although separate temperatures for the

frozen film on the cladding and structure are tracked in PLUTO2, only one common smear density is tracked for the frozen films. This should be improved in a future PLUTO2 version.

When fuel plateout has occurred, the generalized densities of the mobile and frozen fuel, the generalized volume fractions $\theta_{\text{ch,op}}$, θ_{fu}, and θ_{vg} of the open channel, mobile fuel, and the vapor/gas mixture are updated. Moreover, the channel hydraulic diameter is updated:

$$D_{\text{ch}}^{\text{new}} = D_{\text{ch}}^{\text{old}} \cdot \frac{\left(\theta_{\text{ch,op}} - \frac{\Delta\rho'_{\text{fu}}}{\rho_{\text{fu,sol}}}\right)}{\theta_{\text{ch,op}}} \tag{14.4-61}$$

This change in the hydraulic diameter is based only on the changes in the open channel cross-section area because the perimeter of the open channel will remain approximately the same whether a fuel crust is present or not. This is a reasonable approximation for an actual subchannel, but it would not be good for a cylindrical pipe.

Fuel crusts can also be released and added back into the moving fuel field. One apparent reason is that the fuel crust can become re-melted due to fission heating and/or the heat flux from the moving fuel. In the code, the crust release due to remelting is initiated when the fuel crust temperature is half way through the melting band. The fraction CFMELT of the crust released per PLUTO2 time step is calculated from:

$$\text{CFMELT} = \frac{T_{\text{ffcl,i}} - \frac{(T_{\text{fu,liq}}+T_{\text{fu,sol}})}{2}}{T_{\text{fu,liq}} - \frac{(T_{\text{fu,liq}}+T_{\text{fu,sol}})}{2}} \tag{14.4-62}$$

The new generalized smear density of the fuel crust is

$$\rho'_{\text{ff,i}} = \rho'^{\text{old}}_{\text{ff,i}} - \Delta\rho_{\text{ff,i}} \tag{14.4-63}$$

where

$$\Delta\rho'^{\text{old}}_{\text{ff,i}} = \rho'^{\text{old}}_{\text{ff,i}} \cdot \text{CFMELT} \tag{14.4-64}$$

A melting fuel crust is assumed to have a higher temperature at the interface with the gas/vapor mixture than at the interface with the cladding because the cladding cannot be significantly molten since this would have led to crust ablation (see below). Therefore the crust thickness is reduced in this case of crust melting:

$$\text{TKFF} = \text{TKFF}^{\text{old}} \cdot (1 - \text{CFMELT}) \tag{14.4-65}$$

The other reason for releasing a fuel crust in PLUTO2 is the melting of the underlying cladding. The frozen fuel crust in mesh cell i will be released in PLUTO2 when the temperature of the middle cladding node has exceeded the input value TECLRL in cell i. A value in the upper part of the cladding melting band is recommended. The frozen crust is rapidly released in this case according to

$$\Delta\rho'_{\text{ff,i}} = \rho_{\text{fu,sol}}\theta_{\text{ch}}\Delta t_{\text{PL}} \cdot \frac{1}{5\text{ ms}} \tag{14.4-66}$$

This means that a crust which would initially fill the entire channel would be completely released in 5 ms. A crust occupying only a fraction FN_{ff} of the channel would be released in $\text{FN}_{\text{ff}} \cdot 5$ ms. The change in the generalized densities of the fuel crust and also the mobile fuel are calculated with Eq. (14.4-63) and 14.4-61. In this case of frozen crust release, it is likely fuel pieces with the full crust thickness will get released and that only the fraction of the channel perimeter, which is covered by frozen crust, which is CFFFCL, gets reduced:

$$\text{CFFFCL}\,(I) = \text{CFFFCL}\,(I) \cdot \frac{\rho'^{\text{new}}_{\text{ff}}}{\rho'^{\text{old}}_{\text{ff}}} \tag{14.4-67}$$

It is important to note that the treatment of frozen crust release after the underlying cladding has melted and also the disallowing of fuel plateout on molten cladding made the successful L8 post-test analysis with PLUTO2

possible [14-15, 14-12]. In the pre-test analysis, these phenomena were not considered and led to a considerable underprediction of the fuel dispersal [14-43, 14-16]. The current treatment, however, is not yet ideal because the released frozen crust pieces are homogenized with the mobile fuel in the node considered. Ideally, these frozen fuel pieces should be treated in a separate chunk field. Such a treatment is being incorporated in the chunk version of the LEVITATE module. Once this version has been made compatible with the SAS4A release version, one can switch to it via the NCPLEV input parameter. This input parameter specifies the number of axial cladding nodes which have to be fully molten before a switch from PLUTO2 to LEVITATE occurs.

For both the remelting of a crust and the release of a frozen crust, energy and velocity adjustments for the moving fuel are made in the lower section of subroutine PLFREZ. Although these adjustments could also be made later in the fuel energy and momentum equations, it is easier to do it in this subroutine because several local variables that are needed would have to become part of the common blocks. The energy adjustment for the moving fuel is:

$$e_{\text{fu,i}}^{\text{new}} = \frac{e_{\text{fu,i}}^{\text{old}} \rho'_{\text{fu,i}} + e_{\text{fu,rl,i}}^{\text{old}} \Delta\rho'_{\text{ff,i}}}{\rho'_{\text{fu,i}} + \Delta\rho'_{\text{ff,i}}} \tag{14.4-68}$$

where

$$e_{\text{fu,rl}} = \begin{cases} e_{\text{fu,liq}} & \text{for a melting crust} \\ e_{\text{ff}} & \text{for a crust which is released due to clad melting} \end{cases}$$

The calculation of the velocity adjustments for the moving fuel at the lower and upper boundaries of the mesh cell i is as follows:

$$u_{\text{fu,i}}^{\text{new}} = u_{\text{fu,i}}^{\text{old}} \cdot \frac{\rho'_{\text{fu,i-1}} + \rho'_{\text{fu,i}}}{\rho'_{\text{fu,i-1}} + \rho'_{\text{fu,i}} + \Delta\rho'_{\text{ff,i-1}} + \Delta\rho'} \tag{14.4-69}$$

$$u_{\text{fu,i+1}}^{\text{new}} = u_{\text{fu,i+1}}^{\text{old}} \cdot \frac{\rho'_{\text{fu,i}} + \rho'_{\text{fu,i+1}}}{\rho'_{\text{fu,i1}} + \rho'_{\text{fu,i+1}} + \Delta\rho'_{\text{ff,i}} + \Delta\rho'_{\text{ff,i+1}}} \tag{14.4-70}$$

If the release of a fuel crust has taken place, the generalized volume fractions $\theta_{\text{ch,op}}$, θ_{fu}, and θ_{vg} of the open channel, the mobile fuel, and the vapor/gas, respectively, are updated at the end of subroutine PLFREZ. Moreover, the hydraulic diameter of the channel is updated

$$D_{\text{ch}}^{\text{new}} = \frac{D_{\text{ch}}^{\text{old}} \cdot \left(\theta_{\text{ch,op}} - \frac{\Delta\rho'_{\text{ff}}}{\rho_{\text{fu,sol}}}\right)}{\theta_{\text{ch,op}}^{\text{old}}} \tag{14.4-71}$$

This change in the hydraulic diameter is based only on changes in the open channel cross-section area because the perimeter of a real sub-channel will be approximately the same whether a fuel crust is present or has been released.

14.4.3.3 Plated-Out and Moving Fuel Configurations

The different configurations of the plated-out and moving fuel in PLUTO2 are shown in Figure 14.4.1 at the beginning of Section 14.4.1. These configurations are shown in a cylindrical geometry although the areas and wetted perimeters used in the code are based on the actual subchannel values. The outer perimeter of the cylindrical channel shown in Figure 14.4.1 represents mostly cladding but also some structure. The various fuel configurations are of importance for calculating the energy and momentum exchange terms because the interface areas are largely determined by the fuel configurations. The fuel flow regime selection has already been described in Section 14.4.3.1. In the present section, the assumptions about the specific fuel configuration in each flow regime are described. In the code, these are implemented near the front of subroutine PLMISC (**PL**UTO2 **MISC**ELLANEOUS).

In the particulate flow regime [see configuration (a) in Figure 14.4.1], the only assumption is that the particles are uniformly distributed in a numerical cell. With regard to the heat transfer between fuel and liquid sodium, both the liquid sodium film and the sodium droplets are also assumed to be uniformly distributed. The heat-transfer area between fuel and liquid Na per unit of generalized smear volume is assumed to be

$$A'_{\text{fi,N1}} = A'_{\text{Pa}} \cdot \left(\frac{\theta_{\text{N1}}}{\theta_{\text{ch,op}}} \right)^{\text{CIA2}} \tag{14.4-72}$$

where

$$A'_{\text{Pa}} = N'_{\text{Pa}} \cdot 4\pi r_{\text{Pa}}^2$$

$N'_{\text{Pa}} = \frac{\rho'_{\text{fu}}}{\frac{4}{3}\pi r_{\text{Pa}}^3 \rho_{\text{fu,sol}}}$, number of fuel particles in a generalized smear volume

θ_{N1} = the generalized liquid sodium volume fraction which includes the moving sodium droplets and the static sodium film.

CIA2 = input constant (see Eq. (14.4-116)).

The heat-transfer area between fuel particles and the vapor/gas mixture is:

$$A'_{\text{fu,vg}} = N'_{\text{Pa}} 4\pi_{\text{Pa}}^2 \cdot \left[1 - \left(\frac{\theta_{\text{N1}}}{\theta_{\text{ch,op}}} \right)^{\text{CIA2}} \right] \tag{14.4-73}$$

For the continuous flow regimes [see (b) through (e) in Figure 14.4.1], the assumed interaction areas between the various components are considerably more complicated than for the particulate flow regime. Several important Fortran variables describing the different configurations for continuous fuel flow will be explained first.

$$\text{ARCH} = \theta_{\text{ch}} \cdot \frac{\text{AXMX}}{\text{NPIN}} \tag{14.4-74}$$

where ARCH is the coolant channel cross section area associated with one pin. AXMX has already been described in Section 14.4.2.1. NPIN is the number of pins per subassembly. Both AXMX and NPIN are input.

$$\text{ARMF} = \text{ARCH} \cdot \frac{\theta_{\text{fu}}}{\theta_{\text{ch}}} \tag{14.4-75}$$

where ARMF is the cross-sectional area of moving fuel associated with one pin.

$$\text{ARFF} = \text{ARCH} \cdot \frac{(\theta_{\text{ch}} - \theta_{\text{ch,op}})}{\theta_{\text{ch}}} \tag{14.4-76}$$

where ARFF is the cross-sectional area of plated-out fuel which is associated with one pin.

$$\text{PECH} = \left(\text{ARCL}' + \text{ARSR}'\right) \cdot \frac{\text{AXMX}}{\text{NPIN}} \tag{14.4-77}$$

where PECH is the channel perimeter associated with one pin. This also includes a fraction of the structure perimeter. The quantities ARCL' and ARSR' are the cladding and structure surface areas per unit of generalized smear volume. Moreover, they are also the total perimeters of the cladding and structure (times one meter) in a unit of generalized smear volume.

The fraction of the cladding and structure which is covered by either molten or plated-out fuel, CFFUCL, has the value 1.0 in the bubbly flow regime. In the annular flow regime (see b,c, and d in Figure 14.4.1), CFFUCL is taken to be a linear function of the total (moving fuel + fuel crust) fuel volume fraction.

$$\text{CFFUCL} = \frac{\text{ARMF} + \text{ARFF}}{\text{ARCH}} \cdot \frac{1}{\text{CIANIN}} \tag{14.4-78}$$

or if CFFUCL calculated from Eq. (14.4-78) is greater than 1.0:

$$\text{CFFUCL} = 1.0 \tag{14.4-79}$$

The latter value applies to condition d in Figure 14.4.1. The input parameter CIANIN in Eq. (14.4-78) determines the fuel volume fraction above which a complete annular fuel flow will exist (i.e., condition d in Figure 14.4.1). A value of 0.5 is recommended, based on the L8 TREAT test analysis with PLUTO2 [14-15, 14-12].

In the annular flow regimes, the thickness TKFU of the layer containing both moving and plated-out fuel is also important. Its calculation is based on the conservation of fuel volume:

$$\text{TKFU} = \frac{(\text{ARFF} + \text{ARMF})}{(\text{PECH} \cdot \text{CFFUCL})} \tag{14.4-80}$$

The frozen fuel crust, which is important in both the annular and the bubbly flow regimes is also characterized by two variables. These are the frozen fuel crust thickness TKFF and the fraction CFFFCL of the channel perimeter that is covered by frozen fuel crust. Both TKFF and CFFFCL are permanent arrays in the code (which are updated during the time steps), whereas the thickness and coverage fraction of the layer including all the fuel are recalculated every PLUTO2 time step. The calculation of the change in the frozen fuel cross-sectional area due to remelting or release has been described in the previous section. The change of the frozen fuel cross section due to freezing can cause both an increase in the frozen crust thickness TKFF and in the coverage fraction CFFFCL. How much either one increases depends on whether bulk type freezing or conduction-limited freezing [14-44, 14-45, 14-46] is more appropriate. In PLUTO2 the mode of plateout of fuel for the partially annular, totally annular and bubbly flow regime is controlled by an input parameter (CIFUFZ) which allows both extremes or values anywhere between these two extremes. The increase in the crust thickness is calculated from:

$$\Delta\text{TKFF} = \frac{\Delta\text{ARFF}}{\text{CFFFCL} \cdot \text{PECH}} \cdot \frac{H_{\text{fu,ff}} \cdot (T_{\text{fu}} - T_{\text{ff}})}{H_{\text{fu,ff}} \cdot (T_{\text{fu}} - T_{\text{ff}}) + H_{\text{fu,cl}} \cdot (T_{\text{fu}} - T_{\text{cl,os}}) \cdot \text{CIFUFZ}} \tag{14.4-81}$$

where

ΔARFF = the change in the frozen fuel cross section during one PLUTO2 time step.

CFFFCL = the fraction of the channel perimeter covered by plated-out fuel at the beginning of the PLUTO2 time step.

CIFUFZ = an input parameter whose value should be between 0 and 1.0. A value of zero leads to pure bulk-type freezing (i.e., the crust grows till it reaches the thickness of the total molten fuel layer thickness before the coverage fraction CFFFCL increases). A value of 1 leads to a conduction-type freezing. CIFUFZ $= 0$ was used in the L8 analysis [14-15, 14-12] because no other option was available at that time.

and the moving fuel-to-frozen-fuel heat flux per unit temperature difference and per unit of generalized smear volume is given by

$$H_{\text{fu,ff}} = h_{\text{fu,ff}} \cdot \text{CFFFCL} \cdot \text{CFMFFF} \cdot \text{PECH} \tag{14.4-82}$$

where

$h_{\text{fu,ff}}$ = the heat-transfer coefficient between molten fuel and frozen fuel which takes the resistance in the fuel crust into account. It is described in the following section on exchange terms.

$$\text{CFMFFF} = \frac{\text{ARMF}}{(\text{ARMF} + \text{ARFF})} \tag{14.4-83}$$

In Eq. (14.4-82) and Eq. (14.4-83), CFMFFF is the fraction of the frozen crust that is assumed to be covered with moving fuel. In the bubbly flow regime, CFMFFF is always 1.0. The moving fuel-to-cladding heat flux required in Eq. (14.4-81) is given by

$$H_{\text{fu,cl}} = h_{\text{fu,cl}}\,(\text{CFFUCL} - \text{CFFFCL}) \cdot \text{PECH} \tag{14.4-84}$$

where

$h_{\text{fu,cl}}$ is the heat-transfer coefficient between moving fuel and cladding, which is described in the following section on exchange coefficients.

The initialization of the crust thickness during the first time step when plateout commences is also affected by the input parameter CIFUFZ that was discussed above:

$$\text{TKFF} = \frac{\text{ARMF} + \text{ARFF}}{\text{PECH} \cdot \text{CFFUCL}} \cdot (1 - \text{CIFUFZ}) + \frac{\text{ARFF}}{\text{PECH} \cdot \text{CFFUCL}} \cdot \text{CIFUFZ} \tag{14.4-85}$$

where the first term on the right-hand side would lead to the maximum initial crust thickness for CIFUFZ $= 0$ and the second term gives the minimum thickness if CIFUFZ $= 1$. The latter will cause the frozen fuel coverage fraction of the channel perimeter (CFFFCL) to be the same as the one for the total (moving + frozen) fuel, CFFUCL. Since TKFF and CFFFCL are permanent arrays in the code, the initial values of these variables are quite important. The frozen fuel coverage fraction of the channel perimeter is calculated, based on frozen fuel volume conservation:

$$\text{CFFFCL} = \frac{\text{ARFF}}{(\text{TKFF} \cdot \text{PECH})} \tag{14.4-86}$$

If CFFFCL $>$ CFFUCL, CFFFCL and TKFF will be:

$$\text{CFFFCL} = \text{CFFUCL} \tag{14.4-87}$$

and

$$\text{TKFF} = \frac{\text{ARFF}}{(\text{PECH} \cdot \text{CFFUCL})} \tag{14.4-88}$$

The crust dimension calculations described above hold in all continuous flow regimes.

14.4.3.4 Energy and Momentum Exchange Terms in Subroutine PLMISC

Besides calculating the moving and frozen fuel configurations, subroutine PLMISC (**PL**UTO2 **MIS**CELLANEOUS) also calculates many heat-transfer and friction coefficients that are later used in the calculation of the fuel energy equations, the sodium energy equation, and in the channel momentum equation. The latter also requires several drag coefficients which are determined in the subroutine calculating the channel momentum equation, PLMOCO. However, some of the variables needed for the drag coefficients are also calculated in subroutine PLMISC.

Several heat and momentum exchange terms described below are still in a simplified or preliminary form. Additional validation efforts, which are outlined in the SAS4A validation plan [14-10], will be necessary to improve certain heat and momentum exchange terms. However, the reasonably successful analyses of TREAT experiments L8 and H6 [14-15, 14-12, 14-6] may be an indication that all of the more important exchange terms are treated properly.

14.4.3.4.1 Heat-transfer and Momentum Exchange Terms Between Sodium and Cladding, and Sodium and Structure

14.4.3.4.1.1 Near-Liquid Sodium

If $\alpha_{\text{Na}} <$ CIVOID, where CIVOID is an input parameter, liquid single-phase correlations will be used if there is still cladding or structure that is not covered by fuel (i.e., particulate or partially annular fuel flow). The input

parameter CIVOID determines the sodium void fraction below which these correlations will be used. A value of 0.5 is recommended based on the L8 analysis [14-15, 14-12]. The heat-transfer coefficient that is used for low-void fraction sodium flow which has a low Prandtl number of about 0.005, is based on Ref. 14-47.

$$h_{\text{Na,cl,i}} = \left[C1 \cdot \left(D_{\text{N1}} \cdot \rho_{\text{N1}} \cdot |u_{\text{Mi,i}} + u_{\text{Mi,i+1}}| \cdot 0.5 \cdot \frac{C_{\text{p,N1}}}{k_{\text{N1}}} \right)^{C2} + C3 \right] \cdot \frac{k_{\text{N1}}}{D_{\text{ch}}} \cdot \text{CFNACL} \tag{14.4-89}$$

where all variables without axial indexes are calculated at the center of cell i

$C1, C2$ and $C3$ are input constants, and

$k_{\text{N1}} =$ CDNL which is a constant input value for the liquid sodium thermal conductivity

$C_{\text{p,N1}}$ = liquid sodium heat capacity whose temperature dependence is described in the material property section of Chapter 12.

$D_{\text{n1}} = D_{\text{ch}} \cdot \frac{\theta_{\text{N1}}}{\theta_{\text{ch, op}}}$ is the hydraulic diameter for the liquid sodium which is assumed to be in an annular flow regime.

D_{ch} = hydraulic diameter of the open coolant channel which takes the presence of a fuel crust into account (see Eq. (14.4-61))

CFNACL = fraction of the channel perimeter that is wetted by sodium. When the particulate flow regime exists, a fraction of the perimeter can be covered by frozen fuel. The value of CFNACL is in this case

$$\text{CFNACL} = (1 - \text{CFFFCL}) \tag{14.4-90}$$

If an annular flow regime exists

$$\text{CFNACL} = (1 - \text{CFFUCL}) \tag{14.4-91}$$

The CFNACL should actually be lumped together with the heat-transfer area. However, to lump such terms together with heat-transfer coefficients has the advantage that the total heat flow rates, which are calculated at the end of subroutine PLMISC, include only the total cladding or structure areas.

The heat-transfer coefficient between low-void fraction sodium and the structure is set equal to that between low-void fraction sodium and cladding

$$h_{\text{NA,sr,i}} = h_{\text{Na,cl,i}} \tag{14.4-92}$$

The friction factor used for the low-void fraction sodium and fission-gas mixture is:

$$F_{\text{Mi}} = \text{AFR} \cdot (\text{Re}_{\text{Mi}})^{\text{BFR}} \tag{14.4-93}$$

where

AFR, BRF are input constants

Re_{Mi} Reynolds number of the mixture of sodium and fission gas which is calculated from:

$$\text{Re}_{\text{Mi}} = \left\{ D_{\text{ch}} \cdot |u_{\text{Mi,i}} + u_{\text{Mi,i+1}}| \cdot 0.5 \cdot \frac{\rho'_{\text{Mi}}}{(\theta_{\text{vg}} + \theta_{\text{N1}} - \theta_{\text{Na,fm}})} \right\} \Big/ \mu_{\text{Mi}} \tag{14.4-94}$$

where the viscosity of the two-phase mixture is evaluated from a formulation suggested by Dukler [14-48]:

$$\mu_{\text{Mi}} = \frac{\rho'_{\text{Mi}}}{(\theta_{\text{vg}} + \theta_{\text{N1}} - \theta_{\text{Na,fm}})} \cdot \left\{ \frac{x_{\text{Mi}} \mu_{\text{vg}}}{\left(\rho_{\text{Nv}} + \frac{\rho'_{\text{fi}}}{\theta_{\text{vg}}} \right)} + \frac{(1 - x_{\text{Mi}})\, \mu_{\text{N1}}}{\rho_{\text{N1}}} \right\} \tag{14.4-95}$$

where

μ_{Nl} = VINL, viscosity of liquid sodium which is input

μ_{vg} = VIVG, viscosity of the vapor/gas mixture which is input.

$x_{\text{Mi}} = \frac{\left(\theta_{\text{vg}}\rho_{\text{Nv}}+\rho'_{\text{fi}}\right)}{\rho'_{\text{Mi}}}$ quality of moving sodium fission-gas mixture

$\rho'_{\text{Mi}} = \rho'_{\text{Na}} - \rho_{\text{Nl}}\theta_{\text{Na, fm}} + \rho'_{\text{fi}}$

i.e., ρ'_{Mi} is the generalized smear density of the moving sodium and the fission gas.

14.4.3.4.1.2 Liquid Sodium Films Present and a Vapor/Gas Mixture or Two-phase Sodium/Gas Mixture in the Gas Core

If $\theta_{\text{Na, fm}} > 0$ and $\alpha_{\text{Na}} >$ CIVOID,

liquid sodium film evaporation or condensation on the liquid film will take place. A splitting of the currently available liquid film volume fraction, $\theta_{\text{Na, fm}}$, into a film volume fraction for the cladding, $\theta_{\text{cl, fm}}$, and for the structure, $\theta_{\text{sr, fm}}$, is done in the following way:

$$\theta_{\text{sr,fm}} = \text{CINAFO} \cdot \theta_{\text{ch,op}} \cdot \frac{A'_{\text{sr}}}{A'_{\text{sr}} + A'_{\text{cl}}} \tag{14.4-96}$$

$$\theta_{\text{cl,fm}} = \theta_{\text{Na,fm}} - \theta_{\text{sr,fm}} \tag{14.4-97}$$

where CINAFO is an input constant which gives the initial and maximum film volume fraction. However, if $\theta_{\text{cl,fm}} \leq 0$,

$$\theta_{\text{sr,fm}} = \theta_{\text{Na,fm}} \tag{14.4-98}$$

This means, that the structure film has its maximum value as long as a liquid film exists on the cladding. If $T_{\text{Na}} > Tl_{\text{cl,os}}$

$$h_{\text{Na,c1}} = \text{CFNACN} \cdot (1 - \text{CFFFCL}) \tag{14.4-99}$$

where

CFNACN is the sodium condensation coefficient which is input

and for

CFFFCL see Eq. (14.4-81)

If $T_{\text{Na}} < T_{\text{cl,os}}$

$$h_{\text{Na,c1}} = \frac{1}{\left(\frac{1}{\text{CFNAEV}}\right) + \frac{w_{\text{cl,fm}}}{\text{CDNL}}} \cdot (1 - \text{CFFFCL}) \tag{14.4-100}$$

where

CFNAEV is the sodium evaporation coefficient which is input and which should be larger than CFNACN which is discussed above

$w_{\text{cl,fm}}$ is the thickness of the sodium film on the clad

CDNL liquid sodium conductivity which is input

The sodium-to-structure heat-transfer coefficient, $h_{\text{Na,sr}}$, is set equal to the $h_{\text{Na,cl}}$ calculated from Eq. (14.4-100) as long as the structure has a lower temperature than the sodium. In the unlikely case that the structure is hotter

than the sodium, an equation similar to Eq. (14.4-100) is used but with the structure film thickness, $w_{\text{sr,fm}}$, instead of $w_{\text{cl,fm}}$.

The friction factor for the case when liquid sodium films are present is based on Ref. 14-49, page 320, and is calculated in the following way:

$$F_{\text{Mi}} = \text{AFRV} \cdot (\text{Re}_{\text{Mi}})^{\text{BFRV}} \cdot \left\{ 1 + (1 - \text{CFFUCL}) \cdot \frac{300\,(w_{\text{cl,fm}} \cdot A'_{\text{cl}} + w_{\text{sr,fm}} \cdot A'_{\text{sr}})}{D_{\text{ch}} \cdot (A'_{\text{cl}} + A'_{\text{sr}})} \right\} \tag{14.4-101}$$

where

AFRV and BFRV are input constants

Re_{Mi} Reynolds number for the sodium/gas mixture (see Eq. (14.4-94))

14.4.3.4.1.3 No Sodium Films Left and Sodium Temperature Below the Outer Cladding Temperature

This condition is common in the annular fuel flow regime in which liquid sodium films are not allowed. Moreover, in the particulate flow regime, the sodium films may have been completely entrained or evaporated. Two conditions have to be considered. First, there can be a flow of sodium droplets, vapor, and gas in the coolant channels or just a flow of vapor and gas. For void fractions larger than the input value CIA4 (which has to be larger than the input value CIVOID), the heat-transfer coefficient used is a linear interpolation between a convective heat-transfer coefficient for a pure vapor/gas moisture, $h_{\text{vg, cl}}$, and a boiling heat-transfer coefficient HCCLMI which is input. For void fractions smaller than CIA4 and larger than CIVOID, a constant value is used:

$$h_{\text{Na,cl}} = \text{HCCLMI} \cdot \text{CFNACL} \tag{14.4-102}$$

where

CFNACL is the fraction of the channel perimeter which is in contact with the two-phase sodium (see also Eq. (14.4-103))

HCCLMI is an input heat-transfer coefficient describing the forced convection heat transfer between sodium droplets and cladding. HCCLMI should be definitely smaller than the evaporation coefficient CFNAEV and probably also smaller than the condensation coefficient CFNACN. It should also be remembered here that for sodium void fractions less than the input value CIVOID, a single-phase convective heat-transfer coefficient is used for the calculation of $h_{\text{Na, }cl}$ (see Eq. (14.4-89)). Therefore, CIA4 should be larger than CIVOID.

For $\alpha_{\text{Na}} >$ CIA4, the above-mentioned linear interpolation is done:

$$h_{\text{Na,cl}} = \frac{[\text{HCCLMI} \cdot (1 - \alpha_{\text{Na}}) + h_{\text{vg,cl}} \cdot (\alpha_{\text{Na}} - \text{CIA4})] \cdot \text{CFNACL}}{(1 - \text{CIA4})} \tag{14.4-103}$$

where

$\alpha_{\text{Na}} = \frac{(\theta_{\text{ch, op}} - \theta_{\text{N1}} - \theta_{\text{fu}})}{(\theta_{\text{ch, op}} - \theta_{\text{fu}})}$

HCCLMI = input heat-transfer coefficient (see Eq. (14.4-102))

$$\text{CFNACL} = \begin{cases} 1 = \text{CFFFCL} & \text{for the particulate fuel flow regime} \quad \text{(see Eq. 14.4-76)} \\ 1 = \text{CFFUCL} & \text{for the annular fuel flow regime} \quad \text{(see Eq. 14.4-74)} \end{cases}$$

$h_{\text{vg,c1}}$ = convective heat-transfer coefficient between a vapor/gas mixture and cladding which is calculated from a simplified Dittus-Boelter equation in which a Prandtl number of 0.7 is assumed [14-50]:

$$h_{\text{vg,cl}} = 0.02 \cdot \frac{k_{\text{vg}}}{D_{\text{Mi}}} \cdot (\text{Re}_{\text{Mi}})^{0.8} \tag{14.4-104}$$

where

$\mathrm{Re_{Mi}} = \frac{D_{\mathrm{Mi}} \cdot \rho_{\mathrm{vg}} \cdot |u_{\mathrm{Mi,i}} + u_{\mathrm{Mi,i+1}}|}{2\ VIVG}$

VIVG = viscosity for the vapor/gas mixture which is input.

$$\rho_{\mathrm{vg}} = \begin{cases} \frac{\rho'_{\mathrm{Mi}}}{(\theta_{\mathrm{ch,op}} - \theta_{\mathrm{fu}})} & \text{for } \alpha < 1 \\ \frac{\left(\rho_{\mathrm{Nv}} \cdot \theta_{\mathrm{vg}} + \rho'_{\mathrm{fi}}\right)}{(\theta_{\mathrm{ch,op}} - \theta_{\mathrm{fu}})} & \text{for } \alpha = 1 \end{cases}$$

$k_{\mathrm{vg}} =$ CDVG, the input thermal conductivity for a vapor/gas mixture.

$$D_{\mathrm{Mi}} = D_{\mathrm{ch}} \cdot \frac{(\theta_{\mathrm{ch,op}} - \theta_{\mathrm{fu}})}{\theta_{\mathrm{ch,op}}} \tag{14.4-105}$$

For the calculation of the latter quantity, it is assumed that only the cross-sectional area of the open channel is reduced due to a molten fuel film, but not the perimeter.

14.4.3.4.1.4 No Liquid Sodium Films Present and Sodium Temperature Higher Than Cladding Temperature

If $\theta_{\mathrm{cl,fm}} = 0$, $T_{\mathrm{Na}} > T_{\mathrm{cl,os}}$, and $\alpha_{\mathrm{Na}} >$ CIVOID

$$h_{\mathrm{Na,cl}} = \mathrm{CFNACN} \cdot \mathrm{CFNACL} \tag{14.4-106}$$

i.e., the sodium vapor condensation heat transfer is not decreased with increasing void fraction.

Since the calculation of the heat transfer between sodium and cladding is quite complicated, an overview is given in Table 14.4.1.

14.4.3.4.1.5 Heat Transfer Between Sodium and Structure

The calculation of the heat-transfer coefficient between sodium and structure, $h_{\mathrm{Na,sr}}$, is based on the same equations as indicated in Table 14.4.1. This table should be slightly modified by replacing $T_{\mathrm{cl,os}}$ by $T_{\mathrm{sr,os}}$ and by replacing "Na film present on cladding" by "Na film present on structure" in order to make it appropriate as an overview for the sodium-to-structure heat-transfer coefficient.

Heat fluxes per unit of temperature and per unit of generalized smear volume are later needed in the energy equations. They are calculated towards the end of subroutine PLMISC and are simply:

$$H_{\mathrm{Na,cl}} = h_{\mathrm{Na,cl}} \cdot A'_{\mathrm{cl}} \tag{14.4-107}$$

$$H_{\mathrm{Na,sr}} = h_{\mathrm{Na,sr}} \cdot A'_{\mathrm{sr}} \tag{14.4-108}$$

14.4.3.4.1.6 Friction Coefficient When No Liquid Sodium Film is Present

Friction coefficients for calculating the friction on the sodium/gas mixture have already been given for the situation when much liquid sodium is present in the channels Eq. (14.4-93) and for the case when liquid sodium films are still present Eq. (14.4-101).

When no liquid films are present, Eq. (14.4-101) leads to:

$$F_{\mathrm{Mi}} = \mathrm{AFRV} \cdot (\mathrm{Re_{Mi}})^{\mathrm{BFRV}} \tag{14.4-109}$$

This equation is appropriate for the particulate fuel flow regime. In the annular fuel flow regime, the momentum exchange between the moving fuel film and the sodium/gas mixture is included in the drag term, which describes this momentum exchange. Thus, the friction of the mixture due to the interaction with the stationary cladding or fuel crust has to be reduced from that of the whole channel perimeter.

$$F_{\text{Mi}} = \text{AFRV} \cdot (\text{Re}_{\text{Mi}})^{\text{BFRV}} \cdot (1 - \text{CFMFCL} - \text{CFMFFF} \cdot \text{CFFFCL}) \tag{14.4-110}$$

where the quantities in the parentheses (CFMFCL, CFMFFF and CFFFCL) are explained in Eq. (14.4-120), Eq. (14.4-83), and Eq. (14.4-86).

Table 14.4.1: Overview of the Calculation of the Sodium-to-Cladding Heat Transfer

Void Fraction	Equation
$\alpha_{\text{Na}} < \text{CIVOID}$	Eq. (14.4-89) (Single-Phase Correlation)
$\text{CIVOID} \leq \alpha_{\text{Na}} \leq \text{CIA4}$	If liquid Na film present on cladding (only F.R. = 1): For $T_{\text{Na}} > T_{\text{cl,os}}$ Eq. (14.4-99) For $T_{\text{Na}} < T_{\text{cl,os}}$ Eq. (14.4-100) If no liquid Na film left (F.R. = 1 or F.R. = 3): For $T_{\text{Na}} > T_{\text{cl,os}}$ Eq. (14.4-106) For $T_{\text{Na}} < T_{\text{cl,os}}$ Eq. (14.4-102)
$\text{CIA4} < \alpha_{\text{Na}} < 1$	If liquid Na film present on cladding (only F.R. = 1): For $T_{\text{Na}} > T_{\text{cl,os}}$ Eq. (14.4-99) For $T_{\text{Na}} < T_{\text{cl,os}}$ Eq. (14.4-100) If no liquid Na film (F.R. = 1 or F.R. = 3): For $T_{\text{Na}} > T_{\text{cl,os}}$ Eq. (14.4-106) For $T_{\text{Na}} < T_{\text{cl,os}}$ Eq. (14.4-103)
$\alpha_{\text{Na}} = 1$	If $T_{\text{Na}} > T_{\text{cl,os}}$ Eq. (14.4-106) If $T_{\text{Na}} < T_{\text{cl,os}}$ Eq. (14.4-106)

14.4.3.4.2 Fuel-to-Coolant Heat Transfer

14.4.3.4.2.1 Fuel-to-Coolant Heat Transfer in the Particulate Fuel Flow Regime

The heat flow rate per unit of temperature and unit of generalized smear volume is calculated from

$$H_{\text{fu,Na}} = h_{\text{fu,N1}} \cdot A'_{\text{Pa}} \cdot f_{\text{fu,N1}} + h_{\text{fu,vg}} \cdot A'_{\text{Pa}} \, (1 - f_{\text{fu,N1}}) \tag{14.4-111}$$

where $f_{\text{fu,N1}}$ is the fraction of fuel which is in contact with liquid sodium which is discussed below. The A'_{Pa} was discussed in Eq. (14.4-72). The above heat-transfer coefficient between fuel and liquid sodium is based on the original Cho-Wright model which considered only the thermal resistance in the fuel [14-9]:

$$h_{\text{fu,N1}} = \text{CIA1} \cdot \frac{k_{\text{fu}}}{r_{\text{Pa}}} \tag{14.4-112}$$

where

CIA1 is an input constant. A value of 1.0 is recommended based on the L8 and H6 analyses [14-15, 14-12, 14-6]. However, when Eq. (14.4-112) is compared to analytical solutions, a value between 3 and 5 would be appropriate [14-51].

r_{Pa} is the radius of the fuel particles or droplets (see input quantities RAFPLA and RAFPSM)

The heat-transfer coefficient between fuel particles and a vapor/gas mixture is calculated from:

$$\frac{1}{h_{\text{fu,vg}}} = \frac{1}{h_1} + \frac{(r_{\text{Pa}} \cdot 0.1)}{k_{\text{fu}}} \tag{14.4-113}$$

where the heat-transfer coefficient h_1, which determines the heat transfer between the fuel surface and the vapor/gas mixture is based on Ref. 14-22 and a Prandtl number of 0.7

$$h_1 = \frac{k_{\text{vg}}}{2r_{\text{Pa}}} \left[2.0 + 0.54 \cdot \left(\frac{2r_{\text{Pa}} \cdot |u_{\text{Mi}} - u_{\text{fu}}| \cdot \rho_{\text{vg}}}{\mu_{\text{vg}}} \right)^{1/2} \right] \tag{14.4-114}$$

where

μ_{vg} = VIVG, viscosity of the vapor/gas mixture which is input

$\rho_{\text{vg}} = \rho_{\text{Nv}} + \frac{\rho'_{\text{fi}}}{\theta_{\text{vg}}}$

The heat-transfer coefficient h_{fuvg} considers only 1/10 of the possible heat resistance in the fuel particles. This is because the heat capacity of a vapor/gas mixture is so low that only the outer skin of the particles will be affected by the heat loss of the vapor.

The particle surface area per unit of generalized smear volume is

$$A'_{\text{Pa}} = N'_{\text{Pa}} \cdot 4\pi r^2_{\text{Pa}} \tag{14.4-115}$$

N'_{Pa} = number of fuel particles in a generalized smear volume (see Eq. (14.4-72))

The contact fraction between fuel and liquid sodium is calculated in the following way:

$$f_{\text{fu,N1}} = \left(\frac{\theta_{\text{N1}}}{\theta_{\text{ch,op}}} \right)^{\text{CIA2}} \tag{14.4-116}$$

where

CIA2 is an input constant. A value of 2.0 is recommended based on the H6 and L8 analysis [14-15, 14-12, 14-6]. A value of 1.0 appears to be too low, because it implies that the liquid sodium appears to be too low, because it implies that the liquid sodium in a partially voided node is heated at the same rate as if the cell were full of sodium. This is because both the effective fuel surface area and the sodium mass (and thus, the sodium heat capacity) are reduced by the same factor in this case.

When the fuel vapor pressure of the moving fuel is above 10^{-2} MPa, fuel vapor condensation on liquid sodium is considered. The heat flow rate per unit of temperature and unit of generalized smear volume is

$$H_{\text{fv,N1}} = \begin{cases} 0 & \text{if } P_{\text{fv}} < 10^{-2}\text{MPa} \\ \left[\text{CFCOFV} \cdot (f \cdot A'_{\text{cl}} + A'_{\text{sr}}) \cdot \frac{\theta_{\text{vg}}}{\theta_{\text{ch,op}} - \theta_{\text{Na,fm}}} \right] & \text{if } P_{\text{fv}} > 10^{-2} \text{ MPa} \end{cases} \tag{14.4-117}$$

where

P_{fv} = fuel vapor pressure

CFCOFV = fuel vapor condensation coefficient which is input

f = is a multiplier which is zero when $\theta_{\text{fm,cl}} = 0$ and 1 when $\theta_{\text{fm,cl}} > 0$

$h_{\text{fu,Nl}}$ and A'_{Pa} are described in Eq. (14.4-112) and Eq. (14.4-115).

This is a rather simple formulation that is at the one extreme limited by the condensation on liquid sodium and at the other extreme by the heat resistance in the fuel droplets.

14.4.3.4.2.2 Fuel-to-Coolant Heat Transfer in the Annular Fuel Flow Regime

In this fuel flow regime, the contact area between the fuel and the sodium is significantly reduced from that in the particulate regime. The heat flow rate per unit of temperature and per unit of generalized smear volume is calculated from

$$H_{\text{fu,Na}} = A'_{\text{fu}} \cdot \left(\frac{1}{\frac{1}{h_1} + \frac{1}{h_2}} \right) \tag{14.4-118}$$

where

A'_{fu} is the surface area of the molten fuel film per unit of the generalized smear volume which is calculated from:

$$A'_{\text{fu}} = (A'_{\text{cl}} + A'_{\text{sr}}) \cdot (\text{CFMFCL} + \text{CFFFCL} \cdot \text{CFMFFF}) \tag{14.4-119}$$

where

$$\text{CFMFCL} = \text{CFFUCL} - \text{CFFFCL} \tag{14.4-120}$$

i.e., CFMFCL is the fraction of the cladding covered by molten fuel. Regarding CFFUCL, see Eq. (14.4-78); for CFFFCL, see Eq. (14.4-86).

CFMFFF = fraction of the frozen fuel perimeter covered by molten fuel (see Eq. (14.4-83)).

The term h_1 in Eq. (14.4-118) is the heat-transfer coefficient between the bulk of the moving fuel film and the surface of the moving film. It is based on the Deissler correlation [14-23, 14-22] and it is very similar to Eq. (14.4-29), except that a different hydraulic diameter and a different Reynolds number are used (see derivation of Eq. (14.4-30)).

$$h_1 = \frac{1}{D_{\text{fu}}} \cdot \mu_{\text{fu,liq}} \cdot C_{\text{p,fu}} \cdot \text{CIA3} \left(\text{Re}_{\text{fu,Mi}}\right)^{0.8} \tag{14.4-121}$$

where

$\mu_{\text{fu,liq}}$ = liquid fuel viscosity for which the input constant VIFULQ is used

$C_{\text{p,fu}}$ = liquid fuel specific heat for which the input constant CPFU is used

CIA3 = input constant (see Eq. (14.2-36))

$$D_{\text{fu}} = \frac{4 \cdot \text{ARMF}}{(\text{PECH} \cdot \text{CFFUCL})} \tag{14.4-122}$$

where

ARMF = cross-sectional area of moving fuel per pin (see Eq. (14.4-75))

PECH = channel perimeter associated with one pin (see Eq. (14.4-77))

CFFUCL = fraction of the channel perimeter covered by fuel (see Eq. (14.4-78))

$$\text{Re}_{\text{fu,Mi}} = \frac{\rho'_{\text{fu}}}{\theta_{\text{fu}}} \cdot \frac{|u_{\text{Mi,i}} - u_{\text{fu,i}}| + |u_{\text{Mi,i+1}} - u_{\text{fu,i+1}}|}{2} \cdot \frac{D_{\text{fu}}}{\mu_{\text{fu,liq}}} \tag{14.4-123}$$

The value of the heat-transfer coefficient h_2 in Eq. (14.4-118) for sodium void fractions less than the input value CIA4 is

$$h_2 = \text{HCFFMI for } \alpha_{\text{Na}} < \text{CIA4} \tag{14.4-124}$$

where

HCFFMI is an input variable which is the convective heat-transfer coefficient between the moving fuel film surface and the sodium droplet/vapor gas mixture

For sodium void fractions larger than CIA4, an interpolation between the above value and a heat-transfer coefficient between the fuel film and a pure vapor/gas mixture is done similarly to the one in Eq. (14.4-103):

For $\alpha_{\text{Na}} >$ CIA4

$$h_2 = \frac{[\text{HCFFMI} \cdot (1 - \alpha_{\text{Na}}) + h_{\text{vg,fu}} (\alpha_{\text{Na}} - \text{CIA4})]}{(1 - \text{CIA4})} \tag{14.4-125}$$

α_{Na} = sodium void fraction

CIA4 = input void fraction above which the above interpolation 14.4-103 is done. In the L8 and H6 analyses [14-15, 14-12] a value of 0.5 was used for CIA4, because no other option was available at that time. Using a higher CIA4 should boost the pure vapor/gas temperatures which appeared to be on the low side in the L8 and H6 analyses. A higher value may actually lead to a better agreement with the downward voiding observed in the L8 experiment.

$h_{\text{vg,fu}}$ = is the heat transfer coefficient between the vapor/gas mixture and the mobile fuel which is based on the same Dittus-Boelter correlation as used for $h_{\text{vg,cl}}$ (see Eq. (14.4-104)).

$$h_{\text{vg,fu}} = 0.02 \cdot \frac{k_{\text{vg}}}{D_{\text{Mi}}} \cdot (\text{Re}_{\text{vg,fu}})^{0.8} \tag{14.4-126}$$

where

$$\text{Re}_{\text{vg,fu}} = \frac{D_{\text{Mi}} \cdot \rho_{\text{vg}} \cdot |u_{\text{Mi,i}} - u_{\text{fu,i}} + u_{\text{Mi,i+1}} - u_{\text{fu,i+1}}|}{\text{VIVG} \cdot 2} \tag{14.4-127}$$

$$\rho_{\text{vg}} = \begin{cases} \rho_{\text{Nv}} + \frac{\rho'_{\text{fi}}}{\theta_{\text{vg}}} & \text{for } \alpha_{\text{Na}} < 1 \\ \frac{\rho'_{\text{Mi}}}{(\theta_{\text{ch,op}} - \theta_{\text{fu}})} & \text{for } \alpha_{\text{Na}} = 1 \end{cases}$$

VIVG = viscosity of the vapor/gas mixture which is input

D_{Mi} = hydraulic diameter for the mixture flow

Eq. (14.4-126) comes from the Dittus-Boelter equation in which a Prandtl number to the power 0.4 appears [14-50]. Since Prandtl numbers for gases are in a narrow range and since the exponent of the Prandtl number further minimizes the dependency of the heat transfer on Prandtl numbers, an average Prandtl number of 0.686 is used to arrive at Eq. (14.4-127).

The fuel vapor to the sodium/gas mixture heat flow rate term per unit of temperature and unit of smear volume is assumed to be limited by the heat resistance in the molten fuel film:

$$H_{\text{fv,Na}} = \begin{cases} 0 & \text{if } P_{\text{fv}} < 10^{-2} \text{ MPa} \\ h_1 \cdot A'_{\text{fu}} & \text{if } P_{\text{fv}} > 10^{-2} \text{ MPa} \end{cases} \tag{14.4-128}$$

h_1, A'_{fu} are described in Eq. (14.4-118) and Eq. (14.4-119)

14.4.3.4.2.3 Fuel Crust-to-Sodium/Fission-gas Heat Transfer for the Case of Particulate or Annular Fuel Flow Regime

The heat-transfer coefficient between the fuel crust and a two-phase sodium/fission-gas mixture is calculated from

$$\frac{1}{h_{\text{ff,Mi}}} = \frac{1}{h_1} + \frac{\text{TKFF}}{k_{\text{fu}} \cdot 2} \tag{14.4-129}$$

where

TKFF is the frozen fuel crust thickness (see Eq. (14.4-81))

For $\alpha_{\text{Na}} < \text{CIA4},\ h_1 = \text{HCFFMI}$ (see Eq. (14.4-123))

For $\alpha_{\text{Na}} > \text{CIA4},\ h_1$ is based on an interpolation between HCFFMI and a single-phase gas/vapor heat-transfer coefficient (see also Eq. (14.4-125))

$$h_2 = \frac{[\text{HCFFMI} \cdot (1 - \alpha_{\text{Na}}) + h_{\text{vg,os}} (\alpha_{\text{Na}} - \text{CIA4})]}{(1 - \text{CIA4})} \tag{14.4-130}$$

where

$$h_{\text{vg,os}} = 0.02 \frac{k_{\text{vg}}}{D_{\text{Mi,ff}}} (\text{Re}_{\text{vg,ff}})^{0.8} \tag{14.4-131}$$

where

$$D_{\text{Mi,ff}} = \frac{D_{\text{ch}}}{\left(1 - \frac{\theta_{\text{ff}}}{\theta_{\text{cf}}}\right)} - 2 \cdot \text{TKFF} \tag{14.4-132}$$

D_{ch} is hydraulic diameter of the open coolant channel (see Eq. (14.4-61)). By dividing it by the term in the brackets, one gets back to the original hydraulic diameter.

TKFF is the thickness of the frozen fuel crust (see Eq. (14.4-81))

$$\text{Re}_{\text{vg,ff}} = \frac{D_{\text{Mi}} \cdot \rho_{\text{vg}} \cdot |u_{\text{Mi,i}} + u_{\text{Mi,i+1}}|}{\mu_{\text{vg}} \cdot 2} \tag{14.4-133}$$

The latter Reynolds number calculation is very similar to that in Eq. (14.4-127) and several variables are explained there.

The fuel crust-to-sodium/fission-gas heat flow rate per unit temperature and per unit of generalized smear volume is split into a heat transfer to the crust on the cladding and the crust on the structure because temperature are calculated for both crusts.

$$H_{\text{ffcl,Na}} = h_{\text{ff,Mi}} \cdot A'_{\text{cl}} \cdot \text{CFFFCL} \cdot (1 - \text{CFMFFF}) \tag{14.4-134}$$

$$H_{\text{ffsr,Na}} = h_{\text{ff,Mi}} \cdot A'_{\text{sr}} \cdot \text{CFFFCL} \cdot (1 - \text{CFMFFF}) \tag{14.4-135}$$

where

CFFFCL is the fraction of the channel perimeter covered by frozen fuel (see Eq. (14.4-86))

CFMFFF is the fraction of CFFFCL covered by moving fuel, which is zero in the particulate fuel flow regime (see Eq. (14.4-83)).

14.4.3.3.4.2.4 Fuel-to-Sodium/Fission-gas Heat Transfer in the Bubbly Fuel Flow Regime

The description of this heat-transfer process will require some more investigation. On the one hand, the bubble temperatures should quickly adjust to the surrounding fuel temperature. On the other hand, if liquid sodium is entrapped by the bubbly fuel, the achievement of high temperatures will lead to exaggerated sodium vapor pressures that will rapidly disperse fuel. This may be realistic but the premise that significant amounts at liquid sodium can be entrapped by a bubbly fuel flow is still unclear. Sudden fuel dispersal of denser fuel masses such as observed in the SLSF experiment P2 [14-52] or in the TREAT tests L3 and L4 [14-53, 14-54] may have been due to non-prototypical lateral injections of liquid sodium into the molten fuel masses. (However, such lateral injections of liquid sodium may be prototypical in the transition phase.) The current heat-transfer calculation in PLUTO2 is relatively straightforward and can be limited via an input parameter.

For the calculation of the heat transfer between fuel and the sodium/fission-gas mixture in the bubbly fuel flow regime, an estimate of the bubble radius is needed. For the maximum bubble radius, it is assumed that

$$r_{\text{bb,mx}} = D_{\text{Mi}} \cdot 0.5$$

This implies that a string of spherical bubbles is assumed for larger void fractions, rather than one or a few elongated bubbles. For decreasing void fractions, the bubble radius is assumed to be:

$$r_{\text{bb}} = r_{\text{bb,mx}} \cdot \left[0.05 + (1 - 0.05) \cdot \exp\left(-\frac{1 - \text{CIBBIN} - \alpha_{\text{bb}}}{\alpha_{\text{bb}}}\right)\right] \tag{14.4-136}$$

where

CIBBIN is the input void fraction above which a bubbly fuel regime can be initiated.

$$\alpha_{\text{bb}} = \frac{(\theta_{\text{ch,op}} - \theta_{\text{fu}})}{\theta_{\text{ch,op}}}$$

When α_{bb} goes to zero, r_{bb} goes to $r_{\text{bb,mx}} \cdot 0.05$. This is the assumed minimum bubble radius. For the heat-transfer coefficient between bubbly fuel and the two-phase mixture, it is assumed that

$$\frac{1}{h_{\text{fu,Mi}}} = \frac{1}{\text{HCFUBB}} + \frac{r_{\text{bb}}}{k_{\text{Mi}}} \tag{14.4-137}$$

where

HCFUBB is an input heat-transfer coefficient describing the heat transfer between the bulk of the fuel and the bubble surfaces. It is also used to control the heat transfer between fuel vapor and the mixture (see Eq. (14.4-141)).

and

$$k_{\text{Mi}} = \frac{\theta_{\text{N1}} \cdot \text{CDNL} + \theta_{\text{vg}} \cdot \text{CDVG}}{\theta_{\text{N1}} + \theta_{\text{vg}}} \tag{14.4-138}$$

where

CDNL, CDVG are input conductivities for liquid sodium and the vapor/fission-gas mixture, respectively.

The heat flow rate per unit of temperature and unit of generalized smear volume is:

$$H_{\text{fu,Mi}} = A'_{\text{fu,Mi}} \cdot h_{\text{fu,Mi}} \tag{14.4-139}$$

where

$$A'_{\text{fu,Mi}} = \frac{3 \cdot (\theta_{\text{ch,op}} - \theta_{\text{fu}})}{r_{\text{bb}}} \tag{14.4-140}$$

which is the total bubble surface area in a unit of smear volume. $(\theta_{\text{ch,op}} - \theta_{\text{fu}})$ is the total bubble volume in a unit of generalized smear volume.

The heat flow rate term between fuel vapor and the bubbles is assumed to be controlled by the input heat-transfer coefficient HCFUBB

$$H_{\text{fv,Mi}} = \begin{cases} 0 & \text{for } P_{\text{fv}} < 10^{-2} \text{ MPa} \\ \text{HCFUBB} \cdot A'_{\text{fu,Mi}} & \text{for } P_{\text{fv}} > 10^{-2} \text{MPa} \end{cases} \tag{14.4-141}$$

14.4.3.4.3 Moving Fuel-to-Cladding, Moving Fuel-to-Structure, and Moving Fuel-to-Fuel-crust Heat Transfer

In the particulate fuel flow regime, no heat transfer between moving fuel and cladding, structure, or fuel crust is considered. However, fuel-vapor condensation on cladding and structure is considered when there is no liquid sodium film left on the cladding or structure. The heat flow rates per unit of temperature and per unit of generalized smear volume are

$$H_{\text{fv,cl}} = \begin{cases} 0 & \text{if } P_{\text{fv}} < 10^{-2} \text{ MPa} \quad \text{or } \theta_{\text{fm,cl}} = 0 \\ \text{CFCOFV} \cdot A'_{\text{c1}} \cdot \text{CFNACL} & \text{if } P_{\text{fv}} > 10^{-2} \text{ MPa} \quad \text{and } \theta_{\text{fm,cl}} > 0 \end{cases} \tag{14.4-142}$$

$$H_{\text{fv,sr}} = \begin{cases} 0 & \text{if } P_{\text{fv}} < 10^{-2} \text{ MPa} \quad \text{or } \theta_{\text{fm,sr}} = 0 \\ \text{CFCOFV} \cdot A'_{\text{sr}} \cdot \text{CFNACL} & \text{if } P_{\text{fv}} > 10^{-2} \text{ MPa} \quad \text{and } \theta_{\text{fm,sr}} > 0 \end{cases} \tag{14.4-143}$$

where

CFNACL $= 1 -$ CFFFCL (see Eq. (14.4-83))

CFCPOFV = fuel vapor condensation coefficient, which is input.

In the annular and bubbly fuel flow regimes, very similar expressions are used for the heat flow rate terms. The difference lies in the different values for the contact coefficients and the Reynolds numbers that are used.

The heat-transfer coefficient for the moving fuel-to-cladding and moving fuel-to-structure coefficient is the same and it is also used for the calculation of the moving fuel-to-fuel-crust heat transfer. It is based on the Deissler correlation [14-23, 14-22] which was already discussed earlier (see Eq. (14.4-29)). However, a conduction term was added to this correlation because it is not designed for very slow or stagnant flow conditions. For the partial or fully annular fuel flow, the following relationship holds:

$$h_{\text{fu,cl}} = \frac{1}{D_{\text{fu}}} \cdot \text{CIA3} \cdot \text{VIFULQ} \cdot \text{CPFU} \cdot (\text{Re}_{\text{fu}})^{0.8} + 2\frac{k_{\text{fu}}}{\text{TKFU}} \tag{14.4-144}$$

where

TKFU is the thickness of the moving fuel film (see Eq. (14.4-80)). In the above equation, this film thickness is not allowed to become smaller than 1/10 of the channel hydraulic diameter. Thus, the additional term will not dominate the equation except for very slow or stagnant flow conditions.

CIA3, VIFULQ, CPFU are all input constants, which were explained in Eq. (14.4-29).

D_{fu} is the hydraulic diameter of moving fuel film (see Eq. (14.4-121)).

$$\text{Re}_{\text{fu}} = \frac{\rho_{\text{fu}} \cdot D_{\text{fu}} \cdot |u_{\text{fu,i}} + u_{\text{fu,i+1}}|}{\mu_{\text{fu,liq}} \cdot 2} \tag{14.4-145}$$

In the bubbly fuel flow regime, pretty much the same form of the heat-transfer coefficient is used. However, some of the terms are evaluated differently:

$D_{\text{fu}} = D_{\text{ch}}$

TKFU $= \frac{D_{\text{ch}}}{4}$

where D_{cg} is the open channel hydraulic diameter, which takes the frozen crusts into account (see Eq. (14.4-61) and Eq. (14.4-71)).

If the energy of the moving fuel film is below the solidus energy, which is possible if solid fuel gets released from an underlying melting cladding, only the pure conduction part of Eq. (14.4-144) will be used. The thickness TKFU is, in this case, always calculated from Eq. (14.4-80) and no lower limit is assumed for it as in Eq. (14.4-144). The calculation of this special heat-transfer coefficient and of the related heat flow rates are performed near the end of PLMISC, whereas the regular coefficients and flow rates are done in the middle section of PLMISC.

The heat flow rates per unit temperature and per unit of generalized smear volume are

$$H_{\text{fu,cl}} = h_{\text{fu,c}} \cdot A'_{\text{cl}} \cdot \text{CFMFCL} \tag{14.4-146}$$

and

$$H_{\text{fu,sr}} = h_{\text{fu,cl}} \cdot A'_{\text{sr}} \cdot \text{CFMFCL} \tag{14.4-147}$$

where

$$\text{CFMFCL} = 1 - \text{CFFUCL} - \text{CFFFCL} \tag{14.4-148}$$

CFMFCL is the fraction of the channel perimeter covered with moving fuel (see Eq. (14.4-78) and Eq. (14.4-83)). In the case of the bubbly fuel flow regime.

$$\text{CFMFCL} = 1 - \text{CFFFCL} \tag{14.4-149}$$

because all the structure or cladding which is not covered by frozen fuel is assumed to be in contact with moving fuel in this case.

For the fuel vapor condensation on cladding and structure, heat flow rate terms per unit temperature and per unit smear volume are used which are similar to Eq. (14.4-142) and Eq. (14.4-143):

$$H_{\text{fv,cl}} = \begin{cases} 0 & \text{if } P_{\text{fv}} < 10^{-2} \text{ MPa} \\ \text{CFCOFV} \cdot A'_{\text{cl}} \cdot \text{CFNACL} & \text{if } P_{\text{fv}} > 10^{-2} \text{ MPa} \end{cases} \tag{14.4-150}$$

$$H_{\text{fv,cl}} = \begin{cases} 0 & \text{if } P_{\text{fv}} < 10^{-2} \text{ MPa} \\ \text{CFCOFV} \cdot A'_{\text{cl}} \cdot \text{CFNACL} & \text{if } P_{\text{fv}} > 10^{-2} \text{ MPa} \end{cases} \tag{14.4-151}$$

where

CFNACL $= 1 -$ CFFUCL (see Eq. (14.4-78))

The heat-transfer coefficient between moving fuel and a frozen fuel crust is calculated form

$$\frac{1}{h_{\text{hu,ff}}} = \frac{1}{h_{\text{fu,cl}}} + \frac{2 \cdot k_{\text{fu}}}{\text{TKFF}} \tag{14.4-152}$$

where

TKFF = the thickness of the frozen fuel crust (see Eq. (14.4-81) and Eq. (14.4-85)).

$k_{\text{fu}} =$ CDFU which is the input fuel conductivity

The heat flow rates per unit temperature and per unit of smear volume for the annular flow regimes are

$$H_{\text{fu,ffcl}} = h_{\text{fu,ff}} \cdot A'_{\text{cl}} \cdot \text{CFFFCL} \cdot \text{CFMFFF} \tag{14.4-153}$$

$$H_{\text{fu,ffsr}} = h_{\text{fu,ff}} \cdot A'_{\text{sr}} \cdot \text{CFFFCL} \cdot \text{CFMFFF} \tag{14.4-154}$$

where

CFFFCL: see Eq. (14.4-83)

CFMFFF: see Eq. (14.4-134)

In the bubbly fuel flow regime

$$H_{\text{fu,ffcl}} = h_{\text{fu,ff}} \cdot A'_{\text{cl}} \cdot \text{CFFFCL} \tag{14.4-155}$$

$$H_{\text{fu,ffsr}} = h_{\text{fu,ff}} \cdot A'_{\text{sr}} \cdot \text{CFFFCL} \tag{14.4-156}$$

All the frozen fuel is assumed to be covered with moving fuel in the bubbly flow regime. Therefore, CFMFFF is 1.0 in this case.

The multiplication with the A's in the above equations is not done in subroutine PLMISC, but in PLTECS which is called next in the calling sequence.

14.4.3.5 Partial Momentum exchange Terms Between Annular Fuel and the Sodium/Fission-gas Mixture

The friction coefficient CFFRMF for the calculation of the drag between a moving fuel film and the sodium/gas mixture is calculated from

$$\text{CFFRMF} = \text{AFRV} \cdot (\text{Re}_{\text{Mi,fu}})^{\text{BFRV}} \cdot (\text{CFMFCL} + \text{CFFFCL} \cdot \text{CFMFFF}) \tag{14.4-157}$$

where

AFRV, BFRV = input constants

$$\text{Re}_{\text{mi,fu}} = \frac{D_{\text{ch}} \cdot |u_{\text{Mi}} - u_{\text{fu}}| \cdot \rho'_{\text{Mi}}}{\text{VIVG} \cdot (\theta_{\text{vg}} + \theta_{\text{N1}})} \tag{14.4-158}$$

CFMFCL = CFFUCL − CFFFCL

CFFUCL: see Eq. (14.4-78)

CFFFCL: see Eq. (14.4-86)

CFMFFF: see Eq. (14.4-83)

In subroutine PLMISC, a contact coefficient that gives the fraction of the cladding and fuel crust in contact with the moving fuel is also set. It is needed in PLMOCO for the calculation of the fuel friction term. In the annular fuel flow regime, it is:

$$\text{CTFRFU} = \text{CFMFCL} + \text{CFMFFF} \cdot \text{CFFFCL} \tag{14.4-159}$$

The terms on the right-hand side of this equation were also used in Eq. (14.4-157).

If the bubbly fuel flow regime holds:

$$\text{CTFRFU} = 1 \tag{14.4-160}$$

14.4.4 Mobile Fuel and Fuel Crust Energy Equations

14.4.4.1 Mobile Fuel Energy Equation

The mobile fuel energy equation, which is solved at the end of subroutine PLMISC, includes only source and sink terms due to heat transfer. The energy changes due to fuel plateout and the energy changes due to the addition of released or remelting fuel crusts are taken into account in subroutine PLFREZ which was discussed in Section 14.4.3.2. The energy change due to the addition of fuel ejected from the pins is taken into account in subroutine PL1PIN.

The mobile fuel energy equation in differential form reads:

$$\begin{aligned} \frac{\partial}{\partial \mathrm{t}}\left(\rho_{\mathrm{fu}} e_{\mathrm{fu}} A_{\mathrm{fu}}+\rho_{\mathrm{fv}} e_{\mathrm{fv}} A_{\mathrm{vg}}\right)+\frac{\partial}{\partial \mathrm{z}}\left(\rho_{\mathrm{fu}} e_{\mathrm{fu}} u_{\mathrm{fu}} A_{\mathrm{fu}}+\rho_{\mathrm{fv}} e_{\mathrm{fv}} u_{\mathrm{Mi}} A_{\mathrm{vg}}\right) \\ =-\sum_{\mathrm{k}} h_{\mathrm{fu,k}} A_{\mathrm{fu,k}}^{l} \cdot\left(T_{\mathrm{fu}}-T_{\mathrm{k}}\right) \\ -\sum_{\mathrm{l}} h_{\mathrm{fv,l}} A_{\mathrm{fv,l}}^{l} \cdot\left(T_{\mathrm{fu}}-T_{l}\right)+Q \cdot\left(A_{\mathrm{fu}} \rho_{\mathrm{fu}}+A_{\mathrm{vg}} \rho_{\mathrm{fv}}\right) \end{aligned} \tag{14.4-161}$$

where

$A_{\mathrm{fu}}, A_{\mathrm{vg}}$ = cross sectional areas of the mobile liquid or solid fuel and of the vapor/gas mixture, respectively

$A_{\mathrm{fu,k}}^{l}$ = interaction area between moving fuel and component k per unit length

$A_{\mathrm{fv},l}^{l}$ = interaction areas between fuel vapor and component l per unit length

ρ_{fv} = saturated fuel vapor density

ρ_{fu} = theoretical density of liquid or solid fuel

e_{fu} = internal energy of liquid or solid fuel

$$e_{\mathrm{fv}}=e_{\mathrm{fu}}+\lambda_{\mathrm{fv}} \tag{14.4-162}$$

λ_{fv} = fuel heat of vaporization

Q = fission heat source per kg of fuel that is calculated from Eq. (14.4-26).

However, F_{POWER} is always 1 in this case.

By inserting Eq. (14.4-162) into the first term of Eq. (14.4-161), one obtains:

$$\frac{\partial}{\partial \mathrm{t}}()=\frac{\partial}{\partial \mathrm{t}}\left[\left(\rho_{\mathrm{fu}} A_{\mathrm{fu}}+\rho_{\mathrm{fv}} A_{\mathrm{vg}}\right) \cdot e_{\mathrm{fu}}+\rho_{\mathrm{fv}} A_{\mathrm{vg}} \lambda_{\mathrm{fv}}\right] \tag{14.4-163}$$

By inserting Eq. (14.4-163) into Eq. (14.4-161) and dividing the resulting equation by AXMS, and by also using the definitions of the generalized smear densities, one arrives at:

$$\begin{aligned} \frac{\partial}{\partial \mathrm{t}}\left(\rho^{\prime}{}_{\mathrm{fuch}} e_{\mathrm{fu}}\right)+\frac{\partial}{\partial \mathrm{t}}\left(\rho^{\prime}{}_{\mathrm{fv}} \lambda_{\mathrm{fv}}\right)+\frac{\partial}{\partial \mathrm{z}}\left(\rho^{\prime}{}_{\mathrm{fu}} e_{\mathrm{fu}} u_{\mathrm{fu}}+\rho^{\prime}{}_{\mathrm{fv}} e_{\mathrm{fv}} u_{\mathrm{Mi}}\right) \\ =-\sum_{\mathrm{k}} h_{\mathrm{fu,k}} A^{\prime}{}_{\mathrm{fu,k}} \cdot\left(T_{\mathrm{fu}}-T_{\mathrm{k}}\right)-\sum_{l} h_{\mathrm{fv},l} A^{\prime}{}_{\mathrm{fu,k}} \cdot\left(T_{\mathrm{fu}}-T_{l}\right)+Q \rho^{\prime}{}_{\mathrm{fu}} \end{aligned} \tag{14.4-164}$$

where

$\rho^{\prime}{}_{\mathrm{fuch}}=\rho^{\prime}{}_{\mathrm{fu}}=\rho^{\prime}{}_{\mathrm{fv}}$

$\rho^{\prime}{}_{\mathrm{fu}}$ = generalized smear density of the moving liquid or solid fuel

$\rho^{\prime}{}_{\mathrm{fv}}$ = generalized smear density of fuel vapor

$A'_{\text{fu,k}}$ = interaction area between moving liquid or solid fuel and component k in a unit of generalized smear volume

$A'_{\text{fv},l}$ = interaction area between fuel vapor and component l in a unit of generalized smear volume

The different heat flow rates which are summed up in Eq. (14.4-164) can be written as:

$H_{\text{fu,na}} \cdot (T_{\text{fu}} - T_{\text{Ta}})$: see Eq. (14.4-111) for particulate fuel flow and Eq. (14.4-118) for annular flow and Eq. (14.4-139) for bubbly flow

$H_{\text{fu,cl}} \cdot (T_{\text{fu}} - T_{\text{cl,os}})$: see Eq. (14.4-146) and Eq. (14.4-148). For particulate flow this term is zero.

$H_{\text{fu,sr}} \cdot (T_{\text{fu}} - T_{\text{sr,os}})$: see Eq. (14.4-147) and Eq. (14.4-148). For particulate flow this term is also zero.

$H_{\text{fu,ffcl}} \cdot (T_{\text{fu}} - T_{\text{ffcl}})$: see Eq. (14.4-153), Eq. (14.4-155)

$H_{\text{fu,ffsr}} \cdot (T_{\text{fu}} - T_{\text{ffsr}})$: see Eq. (14.4-154), Eq. (14.4-156)

$H_{\text{fv,N1}} \cdot (T_{\text{fu}} - T_{\text{Na}})$: see Eq. (14.4-117), Eq. (14.4-128), and Eq. (14.4-141)

$H_{\text{fv,cl}} \cdot (T_{\text{fu}} - T_{\text{cl,os}})$: see Eq. (14.4-209) and Eq. (14.4-152); in the bubbly flow regime this term is zero

$H_{\text{fv,sr}} \cdot (T_{\text{fu}} - T_{\text{sr,os}})$: see Eq. (14.4-143) and Eq. (14.4-151); in the bubbly flow regime this term is zero

where H is heat transfer coefficient times heat transfer area.

The main part of the energy equation for the moving fuel is solved at the end of subroutine PLMISC. However, heat-transfer terms between moving fuel and the frozen crust on cladding and structure are only included in subroutine PLTECS which is called after PLMISC.

Since the moving fuel energy equation is solved explicitly (i.e., using only beginning-of-time step values in the convective terms and in the heat-transfer terms), it is possible to solve this equation for the fuel energy rather than for fuel temperature. This has the advantage that the heat of fusion can be easily taken into account. The moving fuel energy is stored in a permanent array in PLUTO2, and fuel temperatures, which are needed in the heat-transfer terms and for calculating fuel vapor pressure, are obtained from function subroutine TEFUEG. The equations solved in this subroutine are the following:

If $e_{\text{fu}} < e_{\text{fu,sol}}$

$$C_{\text{p,fu}} = \text{CPFU} \tag{14.4-165}$$

$$T_{\text{fu}} = T_{\text{fu,sol}} - \frac{(e_{\text{fu,sol}} - e_{\text{fu}})}{C_{\text{p,fu}}} \tag{14.4-166}$$

If $e_{\text{fu,sol}} < e_{\text{fu}} < e_{\text{fu,liq}}$

$$C_{\text{p,fu}} = \frac{(e_{\text{fu,liq}} - e_{\text{fu,sol}})}{(T_{\text{fu,liq}} - T_{\text{fu,sol}})} \tag{14.4-167}$$

$$T_{\text{fu}} = T_{\text{fu,sol}} + \frac{(e_{\text{fu}} - e_{\text{fu,sol}})}{C_{\text{p,fu}}} \tag{14.4-168}$$

If $e_{\text{fu,liq}} < e_{\text{fu}}$

$$C_{\text{p,fu}} = \text{CPFU} \tag{14.4-169}$$

$$T_{\text{fu}} = T_{\text{fu,liq}} + \frac{(e_{\text{fu}} - e_{\text{fu,liq}})}{C_{\text{p,fu}}} \tag{14.4-170}$$

The function subroutine TEFUEG is called in subroutine PLSET2, which is called whenever control is transferred to PLUTO2 and which sets all temporary arrays such as the fuel temperature. Moreover, it is called for all nodes containing fuel at the end of subroutine PLTECS after nearly all the updating of the moving fuel energy has

been done. In nodes receiving fuel that is ejected from the pins, the fuel energy is further updated and function subroutine TEFUEG is used again for further updating the fuel temperatures in such nodes.

For the numerical solution of Eq. (14.4-164), the main storage term is rewritten:

$$\frac{\partial}{\partial \mathrm{t}}\left(\rho'_{\mathrm{fuch}} e_{\mathrm{fu}}\right) = \rho'_{\mathrm{fuch}} \cdot \frac{\partial}{\partial \mathrm{t}} e_{\mathrm{fu}} + \frac{\partial \rho'_{\mathrm{fuch}}}{\partial \mathrm{t}} \cdot e_{\mathrm{fu}} \tag{14.4-171}$$

In finite difference form, the above equation is written as

$$\frac{\Delta\left(\rho'_{\mathrm{fuch}} e_{\mathrm{fu}}\right)}{\Delta t} = \rho'^{n+1}_{\mathrm{fuch}} \cdot \frac{\Delta e_{\mathrm{fu}}}{\Delta t} + e^{n}_{\mathrm{fu}} \cdot \frac{\Delta \rho'_{\mathrm{fuch}}}{\Delta t} \tag{14.4-172}$$

which can be arrived at by writing

$$\Delta\left(\rho'_{\mathrm{fuch}} e_{\mathrm{fu}}\right) = \left(\rho'_{\mathrm{fuch}} + \Delta\rho'_{\mathrm{fuch}}\right) \cdot \left(e_{\mathrm{fu}} + \Delta e_{\mathrm{fu}}\right) - \rho'_{\mathrm{fuch}} \cdot e_{\mathrm{fu}}$$

The derivative of the generalized smear density in Eq. (14.4-172) is

$$\frac{\Delta \rho'_{\mathrm{fuch}}}{\Delta t} = -\frac{\Delta\left[\left(\rho'_{\mathrm{fuch}} - \rho'_{\mathrm{fv}}\right) u_{\mathrm{fu}}\right]}{\Delta z} - \frac{\Delta\left(\rho'_{\mathrm{fv}} u_{\mathrm{Mi}}\right)}{\Delta z} \tag{14.4-173}$$

which is the fuel mass conservation equation without source or sink terms. Since the energy Eq. (14.4-164) does not include loss or gain terms due to mass sinks or sources (these are separately included in the fuel freezing and crust release calculations in subroutine PLFREZ and in the fuel ejection calculation in subroutine PLIPIN), the density changes in Eq. (14.4-164) are solely due to mass convection. By writing Eq. (14.4-164) in finite difference form and by including Eq. (14.4-172) and Eq. (14.4-173), one arrives at

$$\begin{aligned} \rho'_{\mathrm{fuch}} \frac{\Delta e_{\mathrm{fu}}}{\Delta t} + \lambda_{\mathrm{fv}} \frac{\partial \rho'_{\mathrm{fv}}}{\partial \mathrm{T}} \frac{\Delta T}{\Delta t} = e_{\mathrm{fu}} \cdot \frac{\Delta\left(\rho'_{\mathrm{pu}} u_{\mathrm{fu}}\right)}{\Delta z} + e_{\mathrm{fu}} \cdot \frac{\Delta\left(\rho'_{\mathrm{pv}} u_{\mathrm{Mi}}\right)}{\Delta z} \\ - \frac{\Delta\left(\rho'_{\mathrm{pu}} e_{\mathrm{fu}} u_{\mathrm{fu}}\right)}{\Delta z} - \frac{\Delta\left(\rho'_{\mathrm{fv}} e_{\mathrm{fv}} u_{\mathrm{Mi}}\right)}{\Delta z} \\ + \sum_{\mathrm{k}} H_{\mathrm{fu,k}} \cdot \left(T_{\mathrm{fu}} - T_{\mathrm{k}}\right) + \sum_{l} H_{\mathrm{fv},l}\left(T_{\mathrm{fu}} - T_{l}\right) + Q' \rho_{\mathrm{fu}} \end{aligned} \tag{14.4-174}$$

In the second term of the left-hand side, it was assumed that the heat of vaporization is constant over the range considered (3700-5000 K). The temperature change in the second term on the left-hand side of Eq. (14.4-174) can be related to the liquid energy change by

$$\frac{\Delta T}{\Delta t} = \frac{\Delta e_{\mathrm{fu}}}{\Delta t \cdot C_{\mathrm{p,fu}}} \tag{14.4-175}$$

By using Eq. (14.4-175) in Eq. (14.4-174), the left-hand side of this equation can be rewritten:

$$\rho'_{\mathrm{fuch}} \frac{\Delta e_{\mathrm{fu}}}{\Delta t} + \lambda_{\mathrm{fu}} \cdot \frac{\partial \rho'_{\mathrm{fv}}}{\partial \mathrm{T}} \frac{\Delta T}{\Delta t} = \frac{\Delta e_{\mathrm{fu}}}{\Delta t}\left(\rho'_{\mathrm{fuch}} + \frac{\lambda_{\mathrm{fv}}}{C_{\mathrm{p,fu}}} \frac{\partial \rho'_{\mathrm{fv}}}{\partial \mathrm{T}}\right) \tag{14.4-176}$$

The convective terms are evaluated by using full donor cell differencing. The first two terms on the right-hand side are actually the convective fluxes of the fuel mass conservation equation (see Eq. (14.4-35)) multiplied by e_{fu} (note that in Eq. (14.4-35), ρ'_{fu} is written as $\rho'_{\mathrm{fuch}} - \rho'_{\mathrm{fv}}$). These convective mass fluxes are used in the energy equation to evaluate the first two terms on the right-hand side of Eq. (14.4-174). The convective energy flux of the solid or liquid fuel is evaluated in the following way (see the schematic below Eq. (14.4-157)).

$$\frac{\Delta\left(\rho'_{\mathrm{fu}} e_{\mathrm{fu}} u_{\mathrm{fu}}\right)}{\Delta z_{\mathrm{i}}} = \frac{\left[\left(\rho'_{\mathrm{fu}} e_{\mathrm{fu}} u_{\mathrm{fu}}\right)_{\mathrm{i+1/2}} - \left(\rho'_{\mathrm{fu}} e_{\mathrm{fu}} u_{\mathrm{fu}}\right)_{\mathrm{i\text{-}1/2}}\right]}{\Delta z_{\mathrm{i}}} \tag{14.4-177}$$

The first term on the right-hand side is evaluated in the following way:

$$\left(\rho'_{\mathrm{fu}} e_{\mathrm{fu}} u_{\mathrm{fu}}\right)_{\mathrm{i+1/2}} = \begin{cases} \rho'_{\mathrm{fu,i}} e_{\mathrm{fu,i}} u_{\mathrm{fu,i+1}} & \text{for } u_{\mathrm{fu,i+1}} > 0 \\ \rho_{\mathrm{fu,i+1}} e_{\mathrm{fu,i+1}} u_{\mathrm{fu,i+1}} & \text{for } u_{\mathrm{fu,i+1}} < 0 \end{cases} \tag{14.4-178}$$

The second term on the right-hand side is evaluated correspondingly. The FORTRAN name for the convective energy fluxes is COFUOS (I). They are calculated in the subroutine solving the mass conservation equations, PLMACO. Eq. (14.4-174) is solved for Δe_{fu} at the end of subroutine PLMISC. As mentioned earlier, the heat flow terms between fuel and the frozen crust on the cladding and on the structure are only later included in subroutine PLTECS, which is called after PLMISC. After these additional updates to the energy equations have been made, the fuel temperature is calculated by using function subroutine TEFUEG (see Eq. (14.4-165)-Eq. (14.4-170)).

14.4.4.2 Fuel Crust Energy Equations

The energy equations for the stationary fuel crusts on the cladding and structure are solved in subroutine PLTECS (**PL**UTO2 **TE**MPERATURE CALCULATION OF **C**LADDING AND **S**TRUCTURE). The calculation of the fuel crust temperatures is done at the beginning of this routine and provides heat flow rates per unit temperature and unit smear volume for the cladding and structure calculation which make up the bulk of this routine and are described in Section 14.5.

Only the temperatures of the fuel crusts on cladding and structure (one for each crust) are stored in permanent arrays in PLUTO2. The beginning of time-step energy of the fuel crust on cladding is determined form the beginning of time-step temperature by

$$e^n_{\text{ffcl}} = \begin{cases} e_{\text{fu,sol}} - (T_{\text{fu,sol}} - T^n_{\text{ffcl}}) \cdot C_{\text{p,fu}} & \text{for } T_{\text{ffcl}} < T_{\text{fu,sol}} \\ \text{EGFUTE}\,(T^n_{\text{ffcl}}) & \text{for } T_{\text{ffcl}} > T_{\text{fu,sol}} \end{cases} \tag{14.4-179}$$

where

$C_{\text{p,fu}} =$ CPFU which is the input value of the fuel specific heat.

EGFUTE is a function subroutine to convert form temperatures to energies. It is could be used over the entire temperature range of the fuel, but by calling it only for the rare case when the crust is above the solidus temperature, computer time is save.

The change in the internal energy of the cladding fuel crust per PLUTO2 time step is calculated from:

$$\begin{aligned} \frac{\Delta e_{\text{ffc1}} \cdot \rho'_{\text{fufm}} \cdot f_{\text{ffcl}}}{\Delta t} &= Q \cdot \rho'_{\text{fufm}} \cdot f_{\text{ffcl}} - H_{\text{ff,Na}} \cdot (T_{\text{ffcl}} - T_{\text{Na}}) \\ &\quad - H_{\text{ff,fu}} \cdot (T_{\text{ffcl}} - T_{\text{fu}}) - H_{\text{ff,cl}} \cdot (T_{\text{ffcl}} - T_{\text{cl,os}}) \end{aligned} \tag{14.4-180}$$

where

$f_{\text{ffcl}} = \frac{A'_{\text{cl}}}{A'_{\text{cl}} + A'_{\text{sr}}}$ is the fraction of all the fuel crust which is on the cladding

$H_{\text{ff,Na}}$ has been described in Eq. (14.4-134).

$H_{\text{ff,fu}}$ has been described in Eq. (14.4-135).

$$H_{\text{ff,cl}} = \frac{k_{\text{fu}} \cdot 2}{\text{TKFF}} \cdot A'_{\text{cl}} \cdot \text{CFFFCL} \tag{14.4-181}$$

where

TKFF is the thickness of the frozen fuel crust

CFFFCL is the fraction of the cladding perimeter covered by fuel crust (see Eq. (14.4-81))

After the new internal energy has been calculated from

$$H_{\text{ff,cl}} = \frac{k_{\text{fu}} \cdot 2}{\text{TKFF}} \cdot A'_{\text{cl}} \cdot \text{CFFFCL} \tag{14.4-182}$$

the new temperature is calculated from

$$T_{\text{ffcl}}^{n+1} = \begin{cases} T_{\text{fu,sol}} - \frac{e_{\text{ffcl}}^{n+1}}{C_{\text{p,fu}}} & \text{for } e_{\text{ffcl}} < e_{\text{fu,sol}} \\ \text{TEFUEG}\left(e_{\text{ffcl}}^{n+1}\right) & \text{for } e_{\text{ffcl}} > e_{\text{fu,sol}} \end{cases} \tag{14.4-183}$$

The function subroutine TEFUEG for converting from fuel energies to temperatures is only called for the rate case when the crust energy is above the solidus energy in order to save computer running time.

The calculation of the structure crust temperature, T_{ffsr}, is done correspondingly. In this case, the fraction of the crust that is on the structure is assumed to be:

$$f_{\text{ffsr}} = \frac{A'_{\text{sr}}}{\left(A'_{\text{sr}} + A'_{\text{cl}}\right)} \tag{14.4-184}$$

14.4.5 Sodium/Fission-gas Energy Equation and Channel Pressure Calculation

The subroutine PLNAEN (**PL**UTO **NA EN**ERGY) which calculates the sodium temperature change for the two-phase sodium/fission-gas mixture and the single phase sodium/vapor/fission-gas mixture is nearly the same as the LEVITATE sodium energy equation (see Section 16.4.3.5) and therefore is not described in detail here. Only the slight differences between the two routines will be discussed.

One difference is due to the fact that LEVITATE treats more components than PLUTO2. These additional components are the moving molten steel films (designated by subscripts **se**) and the moving fuel and steel chunks (designated by subscript **f1** and **s1**, respectively). Because of the presence of these components, three additional heat-transfer terms appear in the LEVITATE sodium energy subroutine LENAEN.

Another difference between the two sodium energy equations is due to the fact that PLUTO2 treats liquid sodium films on the cladding. This has not yet been incorporated in LEVITATE. In PLUTO2, the convective energy fluxes for the two-phase sodium/gas mixture use sodium densities and sodium qualities which exclude the liquid sodium film. The quality of the moving two-phase mixture in PLUTO2 is

$$x_{\text{Mi}} = \frac{\rho_{\text{Nv}} \cdot \theta_{\text{vg}}}{\left(\rho'_{\text{Na}} - \rho'_{\text{Na,fm}}\right)} \tag{14.4-185}$$

In LEVITATE, the sodium quality x_{Na} is also based on the above equation, but with $\rho'_{\text{Na,fm}} = 0$ in all nodes. It should be noted here that the liquid film treatment in PLUTO2 does not come into the energy equation in other ways because the liquid sodium film is considered to be in thermal equilibrium with the moving sodium in a node.

Subroutine PLNAEN includes most of the PLUTO2 channel pressure calculation. However, an updating is done later in subroutine PL1PIN for cells into which fuel and/or gas is injected. Subroutine PLNAEN includes the calculation of fission-gas pressure, sodium saturation pressure or superheated sodium vapor pressure, fuel vapor pressure, and, if necessary, single-liquid-phase pressure of sodium. LEVITATE does not treat the latter because it is not important for voided channel conditions. LEVITATE has a much more refined fuel vapor pressure calculation than PLUTO2, tub this is not performed in the LEVITATE sodium energy equation LENAEN. Moreover, LEVITATE treats steel vapor pressure. But this is also not calculated in LENAEN.

In single-phase liquid sodium pressures do not play a significant role, PLUTO2 will add up the partial pressures according to Dalton's law:

$$P_{\text{ch}} = P_{\text{Nv}}\left(T_{\text{Na}}\right) + P_{\text{fv}}\left(T_{\text{fu}}\right) + P_{\text{fi}}\left(T_{\text{Na}}\rho_{\text{fi}}\right) \tag{14.4-186}$$

In the above equation, the fission-gas pressure contribution is based on the ideal-gas equation (see Eq. (14.4-187) and Eq. (14.4-188)). The fission gas is assumed to be always in thermal equilibrium with the two-phase

sodium mixture. Therefore, no separate fission-gas temperature appears in the equation-of-state. For sodium void fractions greater than 70%, the fission-gas pressure is calculated from the ideal-gas equation:

$$P_{\text{fi}} = \frac{R_{\text{fi}} \rho'_{\text{fi}} T_{\text{Na}}}{\theta_{\text{vg,un}}} \tag{14.4-187}$$

where

R_{fi} = RGAS gas constant for fission gas which is the universal gas constant divided by the averaged molecular weight of xenon, krypton, and helium (the latter is only important for near-fresh fuel)

$\theta_{\text{vg,un}}$ is the generalized volume fraction of the vapor gas space when the compressibility of liquid sodium is not taken into account.

For sodium void fractions of less than 70%, the fission-gas pressure calculation takes the sodium compressibility into account. This is done by solving the ideal-gas equation which includes the sodium compressibility:

$$P_{\text{fi}} = \frac{R_{\text{fi}} \rho'_{\text{fi}} T_{\text{Na}}}{\theta_{\text{vg,un}} + \theta_{\text{N1}} K_{\text{N1}} P_{\text{fi}}}$$

The positive solution of this quadratic equation is

$$P_{\text{fi}} = \frac{-\theta_{\text{vg,un}} + \left[\theta^2_{\text{vg,un}} + 4\theta_{\text{N1}} K_{\text{N1}} R_{\text{fi}} T_{\text{Na}} \rho'_{\text{fi}}\right]^{1/2}}{2 \cdot \theta_{\text{N1}} K_{\text{N1}}} \tag{14.4-188}$$

where

K_{N1} = CMNL, the adiabatic liquid sodium compressibility

θ_{N1} = generalized volume fraction of liquid sodium

This equation will default to a pure liquid phase equation if the fission-gas density is zero and the $\theta_{\text{vg,un}}$ is negative.

When the latter is true, i.e. when the liquid sodium does not fit into the numerical node without compressing it, the fission-gas pressure, which is calculated from Eq. (14.4-188), is compared with the sum of the saturation pressures:

$$\begin{aligned} &\text{If } \theta_{\text{vg,un}} < 0 \text{ and } P_{\text{fi}} > P_{\text{Nv}} + P_{\text{fv}} \text{ ,} \\ &P_{\text{ch}} = P_{\text{fi}} \end{aligned} \tag{14.4-189}$$

$$\begin{aligned} &\text{If } \theta_{\text{vg,un}} < 0 \text{ and } P_{\text{fi}} < P_{\text{Nv}} + P_{\text{fv}} \text{ ,} \\ &P_{\text{ch}} = P_{\text{fi}} + P_{\text{fv}} + P_{\text{Nv}} \end{aligned} \tag{14.4-190}$$

where P_{fi} is calculated from Eq. (14.4-188).

In the pure vapor/gas regime in which no liquid sodium is left (i.e., sodium quality equals one) the channel is calculated from

$$P_{\text{ch}} = P_{\text{Nv}}\left(T_{\text{Na}}, \rho_{\text{Nv}}\right) + P_{\text{fi}}\left(T_{\text{Na}}, \rho_{\text{fi}}\right) + P_{\text{fv}}\left(T_{\text{fu}}\right)$$

where

$$P_{\text{Nv}} = \frac{\text{RGNA} \cdot \rho'_{\text{Na}} T_{\text{Na}}}{\theta_{\text{vg,un}}} \tag{14.4-191}$$

The "gas constant" RGNA is not really a constant but is based on an interpolation between a special "gas constant", which leads to the sodium saturation pressure when inserted into Eq. (14.4-191), and the actual general gas constant divided by the molecular weight of sodium vapor. This is described in more detail in Section 16.4.3.5 in the LEVITATE description.

14.4.6 Momentum Equations in the Coolant Channel

14.4.6.1 Differential Equations for the Sodium/Vapor/Gas/Mixture and for the Moving Solid or Liquid Fuel

14.4.6.1.1 Momentum Equation for the Sodium/Vapor/Gas Mixture

The momentum equation for the mixture of liquid sodium, sodium vapor, fission gas, and fuel vapor is presented in terms of the generalized smear densities, volume fractions and sources and sinks. The step from the more basic equation, which includes the cross sectional areas of each component, to Eq. (14.4-201) below has been omitted because a similar step has been explained earlier for the mass and energy equations; see Eq. (14.4-15) or Eq. (14.4-161). However, it should be remembered that the following equation is written for variable cross section flow. The variable cross section is included in the generalized smear densities and volume fractions.

$$\begin{aligned}\frac{\partial}{\partial t}\left(\rho'_{Mi}u_{Mi}\right)+\frac{\partial}{\partial z}\left(\rho'_{Mi}u_{Mi}^2\right)&\\ =-\theta_{Mi}\frac{\partial P_{ch}}{\partial z}-\rho'_{Mi}g&\\ -\frac{F_{Mi}\rho'_{Mi}}{2D_{Mi}}\cdot u_{Mi}\cdot|u_{Mi}|-f_{drag}\cdot(u_{Mi}-u_{fu})\cdot|u_{Mi}-u_{fu}|&\\ -f_{bb}\rho_{fu,liq}\frac{\theta_{Mi}}{2}\cdot\left[\frac{\partial}{\partial t}(u_{Mi}-u_{fu})+u_{Mi}\frac{\partial}{\partial z}(u_{Mi}-u_{fu})\right]&\\ -S'_{Na,deet}u_{Mi}-S'_{Na,co}u_{Mi}-S'_{fv,co}u_{Mi}&\end{aligned}\tag{14.4-192}$$

where the last three terms are sink terms due to sodium droplet de-entertainment onto the liquid sodium film, and sodium vapor and fuel vapor condensation on clad and structure. Source terms due to fuel evaporation and dissolved fission gas release are disregarded. All other terms will be explained only after the above equation has been modified by inserting the following mass conservation for the moving sodium/vapor/gas mixture:

$$\frac{\partial\rho'_{Mi}}{\partial t}+\frac{\partial}{\partial z}\left(\rho'_{Mi}u_{Mi}\right)=S'_{fi,ej}+S'_{Na,et}-S'_{Na,deet}-S'_{Na,co}-S'_{fv,co}\tag{14.4-193}$$

where the source and sink terms on the right-hand side are due to free fission-gas equation, sodium film entrainment, sodium droplet de-entrainment onto the liquid film and sodium vapor condensation on cladding and structure. Source terms due to sodium evaporation, fuel evaporation and dissolved fission-gas release are not considered.

Eq. (14.4-193) can be inserted into Eq. (14.4-192) if the first term in Eq. (14.4-192) is split. Splitting this term and inserting the mixture mass conservation equation leads to a momentum equation in which the velocity is the dependent variable and not the mass flux. It can be seen later that this is of key importance for the simultaneous solution of the two momentum equations considered. (If mass and momentum equations for both fluids are solved simultaneously as in SIMMER-II [14-19], this splitting of the mass flux will not be necessary).

Splitting the first term in Eq. (14.4-192) and inserting Eq. (14.4-193) leads to

$$\begin{aligned}\rho'_{Mi}\frac{\partial}{\partial t}u_{Mi}=-\frac{\partial}{\partial z}\left(\rho'_{Mi}u_{Mi}^2\right)+u_{Mi}\cdot\frac{\partial}{\partial z}\left(\rho'_{Mi}u_{Mi}\right)&\\ -\theta_{Mi}\frac{\partial P_{ch}}{\partial z}-\rho'_{Mi}g-\frac{F_{Mi}\rho'_{Mi}}{2D_{Mi}}\cdot u_{Mi}\cdot|u_{Mi}|&\\ -f_{drag}\cdot(u_{Mi}-u_{fu})\cdot|u_{Mi}-u_{fu}|&\\ -f_{bb}\rho_{fu,liq}\cdot\frac{\theta_{Mi}}{2}\cdot\left[\frac{\partial}{\partial t}(u_{Mi}-u_{fu})+u_{Mi}\frac{\partial}{\partial z}(u_{Mi}-u_{fu})\right]&\\ -S'_{Na,et}u_{Mi}-S'_{fi,ej}\cdot u_{Mi}&\end{aligned}\tag{14.4-194}$$

where

f_{bb} = is a factor which is zero of the particulate and annular flow regime and has a value of one for the bubbly flow regime. This means that the apparent mass effect is considered only for the bubbly flow regime. This was done because in this latter flow regime accelerating or decelerating low-density bubbles also have to accelerate or decelerate high-density fuel of half the bubble volume (apparent mass effect). This has a significant effect on the slip between the bubbles and the continuous fuel [14-49].

$S'_{\text{Na,et}}$ = the mass of sodium entrained per unit time and per unit of smear volume

$S'_{\text{fi,ej}}$ = the mass of free fission gas being ejected into the channel per unit of time and unit of smear volume

F_{Mi} = the modified friction coefficient which is different for each flow regime:

$$F_{\text{Mi,FR1}} \text{ is calculated in } \begin{cases} \text{Eq. 14.4-82 if } \alpha_{\text{Na}} < \text{CIVOID} \\ \text{Eq. 14.4-86a if liquid sodium films present} \\ \text{Eq. 14.4-91 if no liquid sodium films present} \end{cases}$$

$F_{\text{Mi,FR3}}$ = see Eq. (14.4-110)

$F_{\text{Mi,FR4}} = 0$ because there is no contact between the sodium/gas mixture and the clad or structure in the bubbly flow regime.

F_{drag} = a part of the drag force which is strongly dependent on the flow regime. For the particulate flow regime this factor is [14-49, 14-63]:

$$f_{\text{drag,FR1}} = \rho_{\text{Mi}}\theta_{\text{fu}}\frac{3}{8f_{\text{Pa}}}\left(\frac{\theta_{\text{Mi}}}{\theta_{\text{ch,op}}}\right)^{\text{CIA5}} \cdot C_{\text{drag}} \tag{14.4-195}$$

where

$$\frac{\theta_{\text{fu}}3}{(8r_{\text{Pa}})} = \frac{1}{2}\cdot\pi r_{\text{Pa}}^2\cdot N'_{\text{Pa}}$$

$N'_{\text{Pa}} = \frac{\theta_{\text{fu}}}{\frac{4}{3}\pi r_{\text{Pa}}^3}$, which is the number of fuel particles in a unit of generalized smear volume

CIA5 is an input controlling the drag dependence on the void fraction. A value of -1.7 is recommended based on references [14-49] and [14-63].

$$C_{\text{drag}} = \begin{cases} 0.44 & \text{for } (\text{Re})_{\text{Pa}} > 500 \\ 18.5\cdot(\text{Re}_{\text{pa}})^{-0.6} & \text{for } (\text{Re})_{\text{Pa}} < 500 \end{cases} \tag{14.4-196}$$

where

$$C_{\text{drag}} = \begin{cases} 0.44 & \text{for } (\text{Re})_{\text{Pa}} > 500 \\ 18.5\cdot(\text{Re}_{\text{pa}})^{-0.6} & \text{for } (\text{Re})_{\text{Pa}} < 500 \end{cases} \tag{14.4-197}$$

μ_{Mi} = viscosity of the sodium/gas mixture (see Eq. (14.4-95))

The partial drag term for the annular fuel flow regime is:

$$f_{\text{drag,FR3}} = \frac{\text{CFFRMF}\cdot\rho'_{\text{Mi}}}{(2D_{\text{Mi}})} \tag{14.4-198}$$

where

CFFRMF includes the Reynolds number dependency of the drag and the fraction of the perimeter covered by moving fuel (see Eq. (14.4-157))

$D_{\text{Mi}} = D_{\text{ch}} \cdot \frac{\theta_{\text{Mi}}}{\theta_{\text{ch,op}}}$, where the D_{ch} accounts for the frozen crust. The multiplication with the ratio of volume fractions accounts for the flow cross section reduction of the mixture due to moving fuel and sodium films. The wetted perimeter is assumed to stay unchanged. This is a reasonable assumption for a true subchannel geometry.

The partial drag term for the bubbly flow is similar to that for the particulate flow (see Eq. (14.4-195)) because non-deformable bubbles are assumed in PLUTO2.

$$f_{\text{drag,FR4}} = \theta_{\text{Mi}} \rho_{\text{fu,liq}} \frac{3}{8 r_{\text{bb}}} C_{\text{drag}} \left(\frac{\theta_{\text{fu}}}{\theta_{\text{ch,op}}} \right)^{\text{CIA5}} \tag{14.4-199}$$

where

r_{bb} = bubble radius which cancels out when the drag coefficient (see Eq. (14.4-202)) is later inserted.

C_{drag} = see discussion below and Eq. (14.4-202)

Transient drag coefficients for bubbly flow are not available. Therefore a drag coefficient is used which is based on steady-state experiments. The calculation of a terminal (for steady-state) velocity can be determined by balancing the gravitational and drag forces.

$$\frac{1}{2} C_{\text{drag}} \rho_{\text{fu}} u_{\text{bb},\infty}^2 \cdot \left(\pi r_{\text{bb}}^2 \right) = \left(\rho_{\text{fu}} - \rho_{\text{Mi}} \right) \cdot g \cdot \left(\frac{4}{3} \pi r_{\text{bb}}^3 \right) \tag{14.4-200}$$

A terminal rise velocity obtained from the drift velocity of the bubbles for the churn turbulent regime similar to that suggested by Zuber [14-19] is used:

$$u_{\text{bb},\infty} = 1.53 \frac{\theta_{\text{ch,op}}}{\theta_{\text{fu}}} \left[\frac{\sigma_{\text{fu}} g \left(\rho_{\text{fu}} - \rho_{\text{Mi}} \right)}{\rho_{\text{fu}}^2} \right]^{1/4} \tag{14.4-201}$$

In the formulation of Zuber, this is a drift velocity. For the relatively high fuel fraction assumed in the bubbly fuel flow regime in PLUTO2, the actual bubble velocity and drift velocity are not very different.

After introducing Eq. (14.4-201) into Eq. (14.4-200) one obtains:

$$C_{\text{drag}} = \frac{\sqrt{\left(\rho_{\text{fu}} - \rho_{\text{Mi}} \right) g}}{\sqrt{\sigma_{\text{fu}}}} \cdot C_{\text{x}} \cdot r_{\text{bb}} \left(\frac{\theta_{\text{fu}}}{\theta_{\text{ch,op}}} \right)^2 \tag{14.4-202}$$

where

C_{x} = 1.1392 based on the above Eq. (14.4-200) and Eq. (14.4-201). The input value of CIA6 is $C_{\text{x}} \cdot 3/8$ which equals 0.4272.

The great advantage of Eq. (14.4-202) is that the bubble radius, which is difficult to evaluate, appears in the numerator and therefore cancels in the drag term described in Eq. (14.4-199). However, it is somewhat questionable whether the bubbly flow in PLUTO2 is really in a churn turbulent regime. At any rate, the above formulation gives a rather high draft force, which is appropriate for a bubbly flow regime.

14.4.6.1.2 Momentum Equation for the Moving Liquid or Solid Fuel

The differential equation for the fuel momentum conservation is presented in a modified form in which the fuel mass conservation is already included. The latter applies only to the liquid or solid fuel (compare with Eq. (14.4-16))

$$\frac{\partial \rho'_{\text{fu}}}{\partial \text{t}} = -\frac{\partial \rho'_{\text{fu}} u_{\text{fu}}}{\partial z} + S'_{\text{fu,ej}} \tag{14.4-203}$$

A sink term due to fuel freezing and a source term due to frozen fuel crust release or remelting are not included in this mass conservation and in the momentum conservation because they are calculated separately in the subroutine treating the fuel plateout and crust release (PLFREZ).

By splitting the momentum storage term and by including Eq. (14.4-203) (this is similar to the derivation of Eq. (14.4-194)), one arrives at:

$$\begin{aligned}\rho'_{\text{fu}}\frac{\partial \text{u}_{\text{fu}}}{\partial \text{t}} = &-\frac{\partial}{\partial \text{z}}\left(\rho'_{\text{fu}}u_{\text{fu}}^2\right) + u_{\text{fu}}\cdot\frac{\partial}{\partial \text{z}}\left(\rho'_{\text{fu}}u_{\text{fu}}\right)\\ &-\theta_{\text{fu}}\frac{\partial \text{P}_{\text{ch}}}{\partial \text{z}} - \rho'_{\text{fu}}g - \frac{F_{\text{fu}}\rho'_{\text{fu}}}{2D_{\text{fu}}}u_{\text{fu}}\left|u_{\text{fu}}\right|\\ &+f_{\text{drag}}\cdot\left(u_{\text{Mi}}-u_{\text{fu}}\right)\cdot\left|u_{\text{Mi}}-u_{\text{fu}}\right|\\ +f_{\text{bb}}\rho_{\text{fu,liq}}\cdot\frac{\theta_{\text{Mi}}}{2}&\left[\frac{\partial}{\partial \text{t}}\left(u_{\text{Mi}}-u_{\text{fu}}\right)+u_{\text{Mi}}\frac{\partial}{\partial \text{z}}\left(u_{\text{Mi}}-u_{\text{fu}}\right)\right]\\ &+S'_{\text{fu,ej}}\cdot\left(u_{\text{fuca}}\cdot\text{CIFUMO}-u_{\text{fu}}\right)\end{aligned} \tag{14.4-204}$$

where

drag and apparent mass forces are the same as in the mixture momentum Eq. (14.4-194) except that they act in the opposite direction,

$S'_{\text{fu,ej}}$ is the mass of fuel ejected from the failed pins per unit time per unit of generalized smear volume,

and

CIFUMO is an input value between 0 and 1. It determines how much of the axial momentum of the fuel just behind the cladding rupture is retained when the fuel is ejected into the channel. In the L8 analysis [14-15, 14-12] a value of 1 was used. But it appears that for midplane failures, which are not rapidly expanding axially, a lower value may be more appropriate,

F_{fu} is a part of the fuel friction force between moving fuel and cladding, structure, and frozen fuel and is dependent on the fuel flow regime,

$F_{\text{fu,FR1}} = 0$ because no friction loss between fuel particles or droplets and the walls is assumed. This is based on observations form TREAT and CAMEL out-of-pile experiments in which the particulate fuel easily travels long distances.

$$F_{\text{fu,FR3}} = \begin{cases}\text{CIFRFU}\cdot\text{CTFRFU} & \text{for } \text{Re}_{\text{fu}} > \text{CIREFU}\\ \frac{64}{\text{Re}_{\text{fu}}}\cdot\text{CTFRFU} & \text{for } \text{Re}_{\text{fu}} < \text{CIREFU}\end{cases} \tag{14.4-205}$$

$\text{Re}_{\text{fu}} = \frac{\rho_{\text{fu}}\cdot|u_{\text{fu}}|\cdot D_{\text{fu}}}{\mu_{\text{fu}}}$

$\mu_{\text{fu}} =$ VIFULQ is input viscosity of liquid fuel

$D_{\text{fu}} = \frac{\theta_{\text{fu}}\cdot D_{\text{ch}}}{\theta_{\text{ch,op}}\cdot\text{CTFRFU}}$

CTFRFU is the fraction of open channel perimeter covered by moving fuel (this includes fuel moving over crusts; see Section 14.4.3.5).

CIFRFU and CIREFU are both input parameters which should be set in a manner such that there is not a sudden jump in the friction force at $\text{Re}_{\text{fu}} =$ CIREFU

14.4.6.2 Finite Difference Equations, Simultaneous Solution Approach, and Subroutine PLMOCO

One of the problems of using a staggered grid is that the solution of the momentum equation on the cell edges requires many variables that are defined on the cell centers. In PLUTO2, most of the needed quantities are obtained by using half the sum of the upstream and downstream quantities. Therefore, neighboring cells should have a similar length in PLUTO2, although a length ratio of less than 2 to 1 for neighboring cells is not considered to cause significant inaccuracies. In the code, all variables that are obtained by halving the sum of the upstream and downstream values contain the two-letter sequence BD (for boundary) at the end of the variable name. In the finite difference equations, no special labeling of these variables will be made.

The finite difference forms of the momentum equations are not in conservative form, although both the mass and energy equations are in conservative form. This was prompted by the experience that the calculations were not always stable (in particular for stagnation cells) when using conservative momentum equations. The latter can be easily obtained by not combining the first two terms on the right-hand side of Eq. (14.4-194) and by integrating the equation from the cell midpoint below a cell boundary to the cell midpoint above it (i.e., over a "momentum cell").

In the approach in PLUTO2, the first two terms on the right-hand side of Eq. (14.4-194) are combined:

$$-\frac{\partial}{\partial z}\left(\rho'_{\text{Mi}}u_{\text{Mu}}^2\right)+u_{\text{Mi}}\cdot\frac{\partial}{\partial z}\left(\rho'_{\text{Mi}}u_{\text{Mi}}\right)=-\rho'_{\text{Mi}}u_{\text{Mi}}\frac{\partial}{\partial z}u_{\text{Mi}} \tag{14.4-206}$$

All terms in Eq. (14.4-194) that include a product of velocities are finite differences in the following manner:

$$\frac{F_{\text{Mi}}\rho'_{\text{Mi}}}{2D_{\text{Mi}}}\cdot u_{\text{Mi}}\cdot|u_{\text{Mi}}|=\frac{F_{\text{Mi}}\rho'_{\text{Mi}}}{2D_{\text{Mi}}}\cdot\left(u_{\text{Mi}}^n+\Delta u_{\text{Mi}}\right)\cdot|u_{\text{Mi}}^n| \tag{14.4-207}$$

$$f_{\text{drag}}\cdot\left(u_{\text{Mi}}-u_{\text{fu}}\right)\cdot|u_{\text{Mi}}-u_{\text{fu}}|=f_{\text{drag}}\cdot\left(u_{\text{Mi}}^n+\Delta u_{\text{Mi}}-u_{\text{fu}}^n-\Delta u_{\text{fu}}\right)\cdot|u_{\text{Mi}}^n-u_{\text{fu}}^n| \tag{14.4-208}$$

$$u_{\text{Mi}}\frac{\partial}{\partial z}\left(u_{\text{Mi}}-u_{\text{fu}}\right)=\left(u_{\text{Mi}}^u-\Delta u_{\text{Mi}}\right)\cdot\frac{1}{\Delta z}\delta\left(u_{\text{Mi}}^u-\Delta_{\text{fu}}^n\right) \tag{14.4-209}$$

where Δ implies change over a time step at a given mesh point and δ implies change over a mesh interval at a given time.

By inserting Eq. (14.4-206) through Eq. (14.4-209) into Eq. (14.4-194), by switching from partial derivatives to δ's and Δ's, and by collecting all terms which include Δu_{Mi} on the left-hand side yields:

$$\begin{aligned}\Delta u_{\text{Mi}}\cdot\Bigg\{\frac{\rho'_{\text{Mi}}}{\Delta t}+\frac{F_{\text{Mi}}\rho'_{\text{Mi}}}{2D_{\text{Mi}}}\cdot|u_{\text{Mi}}|+f_{\text{drag}}\cdot|u_{\text{Mi}}-u_{\text{fu}}|+f_{\text{bb}}\cdot\rho_{\text{fu,liq}}\cdot\frac{\theta_{\text{Mi}}}{2}\cdot\\ \left(\frac{1}{\Delta t}+\frac{\delta\left(u_{\text{Mi}}-u_{\text{fu}}\right)}{\Delta z}\right)\Bigg\}=-\rho'_{\text{Mi}}u_{\text{Mi}}\frac{\delta u_{\text{Mi}}}{\Delta z}\\ -\theta_{\text{Mi}}\frac{\Delta P_{\text{ch}}}{\Delta z}-\rho'_{\text{Mi}}g-\frac{F_{\text{Mi}}\rho'_{\text{Mi}}}{2D_{\text{Mi}}}\cdot u_{\text{Mi}}\cdot|u_{\text{Mi}}|\\ -f_{\text{drag}}\cdot\left(u_{\text{Mi}}-u_{\text{fu}}\right)\cdot|u_{\text{Mi}}-u_{\text{fu}}|+f_{\text{drag}}\cdot\Delta u_{\text{fu}}\cdot|u_{\text{Mi}}-u_{\text{fu}}|\\ +f_{\text{bb}}\rho_{\text{fu,liq}}\frac{\theta_{\text{Mi}}}{2}\cdot\left[\frac{\Delta u_{\text{fu}}}{\Delta t}-u_{\text{Mi}}\frac{\delta\left(u_{\text{Mi}}-u_{\text{fu}}\right)}{\Delta z}\right]\\ -S'_{\text{Na,et}}u_{\text{Mi}}-S'_{\text{fi,ej}}u_{\text{Mi}}\end{aligned} \tag{14.4-210}$$

The superscripts n, that denote the beginning of the time step, have been dropped in this equation. Before elaborating on the spatial differencing, the elimination of the Δu_{fu}'s on the right-hand side of this equation will be described. This is achieved by inserting the finite difference form of the fuel momentum equation into the above equation. The finite difference form of the fuel momentum equation can be obtained by performing the

same manipulations which were done to arrive at Eq. (14.4-210). The fuel momentum Eq. (14.4-204) reads in finite difference form:

$$\Delta u_{\text{fu}} = \frac{\text{AHELP}}{\text{BHELP}} + \Delta u_{\text{Mi}} \cdot \left\{ \frac{f_{\text{drag}} \cdot |u_{\text{Mi}} - u_{\text{fu}}|}{\text{BHELP}} + f_{\text{bb}} \rho_{\text{fu,liq}} \frac{\theta_{\text{Mi}}}{2} \cdot \left(\frac{1}{\Delta t} + \frac{\delta\,(u_{\text{Mi}} - u_{\text{fu}})}{\Delta z} \right) \cdot \frac{1}{\text{BHELP}} \right\} \tag{14.4-211}$$

where the second term in the bracket is solely due to the apparent mass (or inertial) force and

$$\begin{aligned} \text{AHELP} = {} & -\rho'_{\text{fu}} u_{\text{fu}} \frac{\delta u_{\text{fu}}}{\Delta z} - \theta_{\text{fu}} \frac{\Delta P_{\text{ch}}}{\Delta z} - \rho'_{\text{fu}} g \\ & - \frac{F_{\text{fu}} \rho'_{\text{fu}}}{2 D_{\text{vu}}} u_{\text{fu}} |u_{\text{fu}}| + f_{\text{drag}} \cdot (u_{\text{Mi}} - u_{\text{fu}}) \cdot |u_{\text{Mi}} - u_{\text{fu}}| \\ & + f_{\text{bb}} \rho_{\text{fu,liq}} \frac{\theta_{\text{Mi}}}{2} \cdot \frac{\delta\,(u_{\text{Mi}} - u_{\text{fu}})}{\Delta z} \cdot u_{\text{Mi}} \\ & + S'_{\text{fu,ej}} \cdot (u_{\text{fuca}} \cdot \text{CIFUMO} - u_{\text{fu}}) \end{aligned} \tag{14.4-212}$$

$$\begin{aligned} \text{BHELP} = {} & \frac{\rho'_{\text{fu}}}{\Delta t} + \frac{F_{\text{fu}} \rho'_{\text{fu}}}{2 D_{\text{vu}}} \cdot |u_{\text{fu}}| + f_{\text{bb}} \rho_{\text{fu,liq}} \frac{\theta_{\text{Mi}}}{2} \cdot \frac{1}{\Delta t} \\ & + f_{\text{drag}} \, |u_{\text{Mi}} - u_{\text{fu}}| \end{aligned} \tag{14.4-213}$$

Eq. (14.4-211) can now be inserted into Eq. (14.4-210) in order to eliminate the Δu_{fu}'s from Eq. (14.4-210). This is necessary to order to perform the simultaneous solution of the mixture and fuel momentum equations. Without this simultaneous solution of the momentum equations, the solution of this two-fluid problem is not stable. The main reason is that the drag terms in the two momentum equations, which act in opposite directions and can be quite large, would not always have the same absolute value, if not solved simultaneously. These discrepancies between the absolute values of the drag would cause serious instabilities.

By inserting Eq. (14.4-211) into Eq. (14.4-210) and collecting all the terms with Δu_{Mi} on the left-hand side, one obtains:

$$\begin{aligned} & \Delta u_{\text{Mi}} \cdot \left\{ \frac{\rho'_{\text{Mi}}}{\Delta t} + \frac{F_{\text{Mi}} \rho'_{\text{Mi}}}{2 D_{\text{Mi}}} \cdot |u_{\text{Mi}}| + f_{\text{drag}} \cdot |u_{\text{Mi}} - u_{\text{fu}}| - f_{\text{drag}}^2 \frac{(u_{\text{Mi}} - u_{\text{fu}})^2}{\text{BHELP}} + \text{BMIIN} \right\} \\ & = -\rho'_{\text{Mi}} u_{\text{Mi}} \frac{\delta u_{\text{Mi}}}{\Delta z} - \theta_{\text{Mi}} \frac{\Delta P_{\text{ch}}}{\Delta z} - \rho'_{\text{Mi}} g - \frac{F_{\text{Mi}} \rho'_{\text{Mi}}}{2 D_{\text{Mi}}} \cdot u_{\text{Mi}} \cdot |u_{\text{Mi}}| \\ & - f_{\text{drag}} \cdot (u_{\text{Mi}} - u_{\text{fu}}) \cdot |u_{\text{Mi}} - u_{\text{fu}}| + f_{\text{drag}} \cdot \frac{\text{AHELP}}{\text{BHELP}} \cdot |u_{\text{Mi}} - u_{\text{fu}}| \\ & + \text{AMIIN} - S'_{\text{Na,et}} u_{\text{Mi}} - S'_{\text{fi,ej}} u_{\text{Mi}} \end{aligned} \tag{14.4-214}$$

where

$$\begin{aligned} \text{AMIIN} = {} & -f_{\text{bb}} \rho_{\text{fu,liq}} \frac{\theta_{\text{Mi}}}{2} u_{\text{Mi}} \cdot \frac{\delta\,(u_{\text{Mi}} - u_{\text{fu}})}{\Delta z} \\ & + f_{\text{bb}} \rho_{\text{fu,liq}} \frac{\theta_{\text{Mi}}}{2} \cdot \frac{\text{AHELP}}{(\text{BHELP} \cdot \Delta t)} \end{aligned} \tag{14.4-215}$$

$$\begin{aligned} \text{BMIIN} = {} & -f_{\text{bb}} \rho_{\text{fu,liq}} \frac{\theta_{\text{Mi}}}{2} \cdot \left(\frac{1}{\Delta t} + \frac{\delta\,(u_{\text{Mi}} - u_{\text{fu}})}{\Delta z} \right) \\ & - f_{\text{bb}} \rho_{\text{fu,liq}} \frac{\theta_{\text{Mi}}}{2} f_{\text{drag}} \cdot |u_{\text{Mi}} - u_{\text{fu}}| \cdot \left(\frac{2}{\Delta t} + \frac{\delta\,(u_{\text{Mi}} - u_{\text{fu}})}{\Delta t} \right) / \text{BHELP} \\ & - f_{\text{bb}}^2 \rho_{\text{fu,liq}}^2 \cdot \frac{\theta_{\text{Mi}}^2}{4} \cdot \frac{1}{\Delta t} \cdot \left(\frac{1}{\Delta t} + \frac{\delta\,(u_{\text{Mi}} - u_{\text{fu}})}{\Delta z} \right) / \text{BHELP} \end{aligned} \tag{14.4-216}$$

AMIIN and BMIIN includes most of the terms related to the apparent mass force (the others are included in AHELP and BHELP).

Eq. (14.4-214) can be solved for Δu_{Mi}. This Δu_{Mi} is then used in the fuel momentum Eq. (14.4-211) to solve for the fuel velocity increment Δu_{fu}. As discussed earlier, this simultaneous solution of the two momentum equations is of key importance for achieving a stable solution of the two-fluid problem.

An item not yet discussed is the finite differencing of the spatial derivatives. As mentioned earlier, an important feature in PLUTO2 is that the convective momentum and mass fluxes are combined (see Eq. (14.4-206)) which makes the momentum equations nonconservative but leads to stable solutions. The spatial differencing of the convective term is demonstrated for the mixture momentum flux term

$$\rho'_{\mathrm{Mi}} u_{\mathrm{Mi}} \frac{\Delta u_{\mathrm{Mi}}}{\Delta z} = \begin{cases} \frac{u_{\mathrm{Mi,i}} \rho'_{\mathrm{Mi,i\text{-}1}} (u_{\mathrm{Mi,i}} - u_{\mathrm{Mi,i\text{-}1}})}{\Delta z_{\mathrm{i\text{-}1}}} & \text{for } u_{\mathrm{Mi,i}} > 0 \\ \frac{u_{\mathrm{Mi,i}} \rho'_{\mathrm{Mi,i}} (u_{\mathrm{Mi,i+1}} - u_{\mathrm{Mi,i}})}{\Delta z_{\mathrm{i}}} & \text{for } u_{\mathrm{Mi,i}} < 0 \end{cases} \tag{14.4-217}$$

The finite differencing of the fuel momentum flux term is done similarly.

The mixture momentum fluxes for the lowermost and for the uppermost cells use the velocities of the sodium slug interfaces if upstream or downstream velocities are needed in the above convective flux calculation. For these nodes, the Δz_{i}'s are the distances between the slug interfaces and the lowermost or uppermost cell boundaries at which the mixture momentum equation is solved.

The fuel velocities at the extremes of the fuel region are needed for the calculation of the convective fuel momentum fluxes at the lowermost and uppermost cell boundaries and also for the calculation of the interface locations of the fuel domain which is done in subroutine PLIF. If the fuel in the uppermost or lowermost fuel node is in a continuous fuel flow regime, the velocity of the upper or lower fuel interface will be set to the velocity of the nearest cell boundary for which the fuel velocity has been calculated by the fuel momentum equations. If the uppermost or lowermost fuel node is in a particulate flow regime, a Lagrangian momentum equation will be solved for a fuel particle at the upper or lower end of the fuel region. The force terms in this momentum equation are equivalent to those in the regular Eulerian momentum equation (see Eq. (14.4-204)), but there is, of course, no convective flux terms in this Lagrangian momentum equation.

Also needed is a spatial differencing of the gradients of the relative velocities between mixture and fuel which appear in Eq. (14.4-215) and 14.4-184. The following upwind differencing which is keyed on the much more sensitive mixture velocity is done in the following way:

$$\frac{\Delta \left(u_{\mathrm{Mi}} - u_{\mathrm{fu}} \right)}{\Delta z} = \begin{cases} \frac{(u_{\mathrm{Mi,i}} - u_{\mathrm{Mi,i\text{-}1}} - u_{\mathrm{fu,i}} + u_{\mathrm{fu,i\text{-}1}})}{\Delta z_{\mathrm{i\text{-}1}}} & \text{for } u_{\mathrm{Mi,i}} > 0 \\ \frac{(u_{\mathrm{Mi,i+1}} - u_{\mathrm{Mi,i}} - u_{\mathrm{fu,i+1}} + u_{\mathrm{fu,i}})}{\Delta z_{\mathrm{i}}} & \text{for } u_{\mathrm{Mi,i}} < 0 \end{cases} \tag{14.4-218}$$

Subroutine PLMOCO (**PL**UTO2 **MO**MENTUM **CO**NSERVATION) sets up all of the coefficients needed for the solution of the momentum conservation. (This is mainly because these coefficients are needed at the cell edges and have previously been set only at the midpoints. However, most interfacial drag terms were not set up earlier.) Subroutine PLMOCO also sets up the convective momentum flux terms and it performs the simultaneous solution fo the two momentum equations. Moreover, the calculation of the fuel region interface velocities, which was described in the previous paragraph, is done. Subroutine PLMOCO also calculates the velocity changes of the liquid sodium slugs above and below the interaction region. This is described in the next section.

14.4.6.3 Velocity Calculation for the Liquid Sodium Slug Interfaces

Ideally, the liquid sodium slugs should be modeled with a fully compressible treatment. Although this was done in the original PLUTO code [14-3. 14-4], it has not been incorporated in PLUTO2 because it is not considered important for whole-core calculations and because a fully compressible calculation in the liquid slugs requires the use of very small time steps. In an earlier stand-alone version of PLUTO2, an optional compressible treatment, which allows a separate time step in the liquid slugs and in the interaction region, should eventually be incorporated into SAS4A/PLUTO2 for use in expensive analyses.

In the currently available treatment in SAS4A/PLUTO2, an acoustic approach is used in the lower and upper slug until the initial pressure waves reach the subassembly inlet and exit, respectively. From then on, the liquid sodium slugs are treated with an incompressible approach.

The initial acoustic approach in PLUTO2 is only used until the pressure wave hits the nearest free surface and not during the round trip time of the expanding and receding pressure wave. The latter is commonly used for the acoustic approximation, but was not used in PLUTO2 because comparisons with the fully compressible PLUTO cod [14.3, 14-4] showed better agreement when only the time for reaching the nearest free surface was used. This time is evaluated in PLUTO2 based on the velocity of sound. The velocity of sound in the lower slug for temperature-independent density and compressibility is:

$$u_{\text{sonic,N1,ls}} = \sqrt{\frac{1}{(\rho_{\text{N1,ls}} \cdot K_{\text{N1}})}} \tag{14.4-219}$$

where

$\rho_{\text{N1,ls}}$ is the average density of the lower sodium slug at the time of PLUTO2 initialization

K_{N1} is the liquid sodium compressibility which is an input constant (see CMNL)

The time to reach the free surfaces at the inlet or outlet are calculated from:

$$\Delta t_{\text{ac,ls}} = \frac{L_{\text{ls}}}{u_{\text{sonic,N1,ls}}} \tag{14.4-220}$$

$$\Delta t_{\text{ac,us}} = \frac{L_{\text{us}}}{u_{\text{sonic,N1,us}}} \tag{14.4-221}$$

where

L_{ls} and L_{us} are the lengths of the lower and upper slug, respectively.

The calculation of the velocities of the interfaces between liquid slugs and interaction region is based on the basic physics equation stating that force is equal to the rate of momentum change. This is applied to a shock front crossing the interface that is driven by a pressure difference ΔP:

$$\frac{\Delta\left(M \cdot u_{\text{if}}\right)}{\Delta t} = -\Delta P \cdot A_{\text{ch}} \tag{14.4-222}$$

where

$$\Delta\left(M \cdot u_{\text{if}}\right) = \rho_{\text{N1}} \cdot A_{\text{ch}} \cdot u_{\text{sonic,N1}} \cdot \Delta t \cdot \left[u_{\text{if}} - u_{\text{if}}\left(t_{\text{o}}\right)\right] \tag{14.4-223}$$

For this equation, the mass accelerated to velocity u_{if} per Δt is the mass which was crossed by the shock wave during Δt. This mass is accelerated through a velocity increment of $\left[u_{\text{if}} - u_{\text{if}}\left(t_{\text{o}}\right)\right]$ due to the crossing of the shock. By inserting Eq. (14.4-223) into Eq. (14.4-222), one obtains:

$$u_{\text{if}} = u_{\text{if}}\left(t_{\text{o}}\right) - \frac{\Delta P}{\rho_{\text{N1}} u_{\text{sonic,N1}}} \tag{14.4-224}$$

For $u_{\text{if}}(t_o) = 0$, this would be equal to the first Rankine-Hugoniot condition for shock waves [14-56] if the actual shock velocity rather than the sonic limit were used.

The above equation is used for the velocity calculation of the lower slug interface during the acoustic period (see Eq. (14.4-220)):

$$u_{\text{if,1s}}(t) = u_{\text{if,ls}}(t_o) + \frac{P_{\text{if,ls}}(t) - P_{\text{inlet}}}{\sqrt{\rho_{\text{N1,ls}}/K_{\text{N1}}}} \tag{14.4-225}$$

In this equation, the definition of the velocity of sound Eq. (14.4-219) and time-dependent interface pressure were introduced. To use a time-dependent pressure is not in the spirit of Eq. (14.4-224) which assumes a constant pressure difference. This represents the main assumption of the acoustic approximation in PLUTO2. Comparison calculations with the fully compressible PLUTO code [14.3, 14-4] have shown that it is a reasonable assumption. For the velocity calculation of the upper slug, the following is used during the acoustic period:

$$u_{\text{if,us}}(t) = u_{\text{if,us}}(t_o) + \frac{P_{\text{if,us}}(t) - P_{\text{outlet}}}{\sqrt{\rho_{\text{N1,us}}/K_{\text{N1}}}} \tag{14.4-226}$$

After the acoustic period for the lower slug is over (see Eq. (14.4-220)), the incompressible calculation of the lower slug mass flow rate begins. For the upper slug, this calculation starts after the time calculated by Eq. (14.4-221) has been exceeded.

A separate incompressible slug calculation is done for the lower sodium slug (below the interaction region) and for the upper slug (above the interaction region). These slugs can occupy several axial channel zones. Each channel zone is characterized by its input flow cross section, hydraulic diameter and axial length. The sodium slugs can fully extend over several channel zones, but the uppermost segment of the lower slug and the lowermost segment of the upper slug do not fully cover a channel zone because of the presence of the interaction region. The length of the uppermost segment of the lower slug and the lowermost segment of the upper slug can vary because they have moving boundaries. A control volume approach for the momentum balance of one slug segment yields (after taking into account the assumption of a constant density in the entire lower or upper slug):

$$\begin{aligned} \rho_{\text{N1,i}} A_{\text{ch,i}} \frac{\Delta\left(u_{\text{N1,i}} L_{\text{i}}\right)}{\Delta t} &= -\rho_{\text{N1}} u_{\text{N1,i+1}}^2 A_{\text{ch,i}} F_{\text{i+1}} + \rho_{\text{N1}} u_{\text{N1,i}}^2, A_{\text{ch,i}} F_{\text{i}} \\ -A_{\text{ch,i}} P_{\text{ch,i+1}} + A_{\text{ch,i}} P_{\text{ch,i}} - f_{\text{fr,i}} \cdot L_{\text{i}} \cdot \frac{\left(u_{\text{N1}} |u_{\text{N1}}| \rho_{\text{N1}} A_{\text{ch}}\right)_{\text{i}}}{2 D_{\text{ch,i}}} &- g L_{\text{i}} \rho_{\text{N1,i}} A_{\text{ch,i}} \\ -0.5 A_{\text{ch,i}} \Delta P_{\text{z=z}_{\text{i}}} - 0.5 A_{\text{ch,i}} \Delta P_{\text{z=z}_{\text{i+1}}} &- A_{\text{ch,i}} \Delta P_{\text{or}} \end{aligned} \tag{14.4-227}$$

where the last three terms describe entrance and exit losses and the losses due to grid spacers in this slug segment and

L_{i} = length of slug segment i

$$\rho_{\text{N1}} = \begin{cases} \rho_{\text{N1,ls}} = & \text{average sodium density for lower slug calculation} \\ & \text{given by the input parameter RHSLBT} \\ \rho_{\text{N1,us}} = & \text{average sodium density for lower slug calculation} \\ & \text{given by the input parameter RHSLBP} \end{cases}$$

$A_{\text{ch,i}}$ = cross section of slug segment i

$F_{\text{i}} = 1$ except i designates the lower (moving) interface of the upper slug. In this case, $F_{\text{i}} = 0$.

$F_{\text{i+1}} = 1$ except when $i+1$ designates the upper (moving) interface of the lower slug. In this case, $F_{\text{i+1}} = 0$.

The left-hand side of Eq. (14.4-227) can be rewritten as

$$\rho_{\text{N1}} A_{\text{ch,i}} \cdot \frac{\Delta\left(u_{\text{N1,i}} L_{\text{i}}\right)}{\Delta t} = \rho_{\text{N1,i}} A_{\text{ch,i}} \left(L_{\text{i}} \frac{\Delta u_{\text{N1,i}}}{\Delta t} + u_{\text{N1,i}} \frac{\Delta L_{\text{i}}}{\Delta t} \right) \tag{14.4-228}$$

The $\frac{\Delta L_i}{\Delta t}$ is only non-zero for the uppermost segment of the lower slug and the lowermost segment of the upper slug. For the case of these two special slug segments:

$$\frac{\Delta L_{\mathrm{i}}}{\Delta t} = u_{\mathrm{N1,i}} \ , \quad \text{for the uppermost segment of the lower slug} \tag{14.4-229}$$

$$\frac{\Delta L_{\mathrm{i}}}{\Delta t} = -u_{\mathrm{N1,i}} \ , \quad \text{for the lowermost segment of the upper slug} \tag{14.4-230}$$

For slug segments that are neither the uppermost one of the lower slug nor the lowermost one of the upper slug, the two convective terms in Eq. (14.4-227) cancel because the velocity is the same everywhere in a slug segment. For the uppermost segment of the lower slug, only the second convective term is present in Eq. (14.4-227) and this one cancels with the last term of the right-hand side of Eq. (14.4-228). For the lowermost segment of the upper slug, only the first convective term is present in Eq. (14.4-227). This one cancels also with the last term in Eq. (14.4-228) because of Eq. (14.4-230), which holds in this case.

After inserting Eq. (14.4-228) into Eq. (14.4-227), cancelling the convective terms and the terms with $\frac{\Delta L_i}{\Delta t}$ (see above discussion), and after dividing the new equation with $A_{\mathrm{ch,i}}$, all the segment equations for each of the two slugs are added up. For the lower slug, this lead to:

$$\begin{aligned} \frac{\Delta W_{\mathrm{ls}}}{\Delta t} \cdot \sum_{\mathrm{i=1}}^{i=\mathrm{IB}} \frac{L_{\mathrm{i}}}{A_{\mathrm{ch,i}}} &= -P_{\mathrm{ch,IB}} + P_{\mathrm{ch,inlet}} \\ - \sum_{\mathrm{i=1}}^{i=\mathrm{IB}} f_{\mathrm{i}} L_{\mathrm{i}} \rho_{\mathrm{N1,ls}} \frac{u_{\mathrm{i}} \left| u_{\mathrm{i}} \right|}{2 D_{\mathrm{ch,i}}} &- \sum_{\mathrm{i=1}}^{i=\mathrm{IB}} L_{\mathrm{i}} \rho_{\mathrm{N1,ls}} g \\ &- \sum_{\mathrm{i=1}}^{i=\mathrm{IB}} \Delta P_{\mathrm{zi}} + \Delta P_{\mathrm{or,ls}} \end{aligned} \tag{14.4-231}$$

where

$W_{\mathrm{ls}} = \rho_{\mathrm{N1,ls}} A_{\mathrm{ch,i}} u_{\mathrm{N1,i}}$, the mass flow rate which is the same in all segments of the lower slug because of the assumed incompressibility

IB = index of the uppermost segment of the lower slug

$P_{\mathrm{ch,IB}}$ = pressure in the first node of the interaction region

ΔP_{zi} = pressure drop due to the area change between segment $i-1$ and segment i or due to an orifice at the bottom of segment i. This pressure drop is evaluated from

$$\Delta P_{\mathrm{zi}} = \mathrm{XKORV}_{\mathrm{i,m}} u_{\mathrm{N1,i}} \left| u_{\mathrm{N1,i}} \right| \frac{\rho_{\mathrm{N1,ls}}}{2} \tag{14.4-232}$$

where $\mathrm{XKORV}_{\mathrm{i,m}}$ is the input contraction or expansion coefficient for upward or downward flow. This can also be an orifice coefficient for segment i. Coefficients with $m = 1$ are used for upward flow; coefficients with $m = 2$ for downward flow, $\Delta P_{\mathrm{or,ls}}$ is the pressure drop due to grid spacers in the channel zone KZPIN and is evaluated from the equation

$$\Delta P_{\mathrm{or,ls}} = N_{\mathrm{or}} \cdot \mathrm{XKORGD} \cdot u_{\mathrm{N1,i}} \left| u_{\mathrm{N1,i}} \right| \frac{\rho_{\mathrm{N1,ls}}}{2} \tag{14.4-233}$$

where

KZPIN = the channel zone which contains the pins

XKORGD = an input pressure-drop coefficient for a single grid spacer

$$N_{\mathrm{or}} = \frac{\mathrm{NGRDSP} \cdot \left(z_{\mathrm{ls,if}} - z_{\mathrm{i=KZPIN}} \right)}{L_{\mathrm{i=KZPIN}}} \tag{14.4-234}$$

where

NGRDSP = the number of uniformly distributed grid spacers in the channel zone KZPIN which is input.

$Z_{\text{ls,if}}$ = axial location of the lower slug interface

$Z_{\text{i=KZPIN}}$ = location of the lower boundary of channel zone KZPIN

For the upper sodium slug, the incompressible momentum equation is

$$\begin{aligned} \frac{\Delta W_{\text{us}}}{\Delta t} \cdot \sum_{\text{i=IT}}^{i+\text{IMAX}} \frac{L_{\text{i}}}{A_{\text{ch,i}}} &= P_{\text{ch,IT}} + P_{\text{ch,outlet}} \\ - \sum_{\text{i=IT}}^{i=\text{IMAX}} f_{\text{i}} L_{\text{i}} \rho_{\text{NI,us}} \frac{u_{\text{i}} \left| u_{\text{i}} \right|}{2 D_{\text{ch,i}}} - \sum_{\text{i+IT}}^{i=\text{MAX}} L_{\text{i}} \rho_{\text{N1,us}} g & \\ - \sum_{\text{i=IT}}^{i-\text{IMAX}} \Delta P_{\text{zi}} + \Delta P_{\text{or,us}} & \end{aligned} \tag{14.4-235}$$

where

IT = index of the lowermost segment of the upper slug

IMAX = index of the top segment of the upper slug

The reason for performing the slug momentum calculations is to determine the upper and lower interface velocities of the interaction region:

$$u_{\text{if,ls}}^{n+1} = u_{\text{if,ls}}^{n} + \Delta W_{\text{ls}} \cdot \frac{1}{\rho_{\text{N1,ls}} A_{\text{IB}}} \tag{14.4-236}$$

and

$$u_{\text{if,us}}^{n+1} = u_{\text{if,us}}^{n} + \Delta W_{\text{us}} \cdot \frac{1}{\rho_{\text{N1,us}} A_{\text{IT}}} \tag{14.4-237}$$

where

A_{IB} = the flow area of the uppermost segment of the lower slug

A_{IT} = the flow area of the lowermost segment of the upper slug.

14.4.6.4 PLUTO2 Time Step Determination

The PLUTO2 time step Δt_{PL} used in the numerical marching of all the in-pin and channel conservation equations is restricted by the sonic Courant conditions for both the in-pin and the channel flows. An upper limit $\Delta t_{\text{PL,pin}}$ of the PLUTO2 time step computed, based on the sonic Courant condition for the in-pin flow of the fuel and fission gas two-phase mixture, is given in Section 14.2.8. In the present section, another upper limit $\Delta t_{\text{PL,ch}}$ of the PLUTO2 time step is computed based on the sonic Courant condition for the multi-component channel flow, and then the smaller of the two upper limits gives the PLUTO2 time step Δt_{PL}. The upper limit $\Delta t_{\text{PL,ch}}$ is computed to be a fraction, 0.4, (same as that used in computing $\Delta t_{\text{PL,pin}}$) of the minimum time step based on the sonic Courant condition for the channel flow.

$$\Delta t_{\text{PL,ch}} + 0.4 \cdot \min \left[\frac{z_1}{(V_{\text{sonic,I}} + |u_{\text{MI,I}}|)} \right]_{\text{I=IFMIBT,IFMITP}} \tag{14.4-238}$$

The minimum in Eq. (14.4-238) is evaluated over all axial cells of the interaction region. The sonic velocity in the channel is calculated from an equation [14-28] for an adiabatic homogeneous two-phase mixture of liquid

sodium and fission gas/sodium vapor. The compressibility of liquid fuel being much smaller than that of liquid sodium, the fuel is assumed to be incompressible in the calculation of the sonic velocity in the channel. The effect of fuel vapor is also not included.

$$V_{\text{sonic}}^2 = \gamma_{\text{vg}} \left(P_{\text{fi}} + P_{\text{Nv}}\right) / \left\{\alpha_{\text{vg}}^2 \left(\rho_{\text{fi}} + \rho_{\text{Nv}}\right) + \alpha_{\text{vg}} \left(1 - \alpha_{\text{vg}}\right) \rho_{\text{N1}} + \left[\alpha_{\text{vg}} \left(1 - \alpha_{\text{vg}}\right) \left(\rho_{\text{fi}} + \rho_{\text{Nv}}\right) + \left(1 - \alpha_{\text{vg}}\right)^2 \rho_{\text{N1}}\right] \gamma_{\text{vg}} \left(P_{\text{fi}} + P_{\text{NV}}\right) K_{\text{N1}}\right\} \tag{14.4-239}$$

where

$\alpha_{\text{vg}} = \frac{\theta_{\text{vg}}}{(\theta_{\text{ch,op}} - \theta_{\text{fu}})}$ = void fraction in the two-phase mixture of liquid sodium and fission gas/sodium vapor.

γ_{vg} = ratio of specific heat at constant pressure to that at constant volume of the fission gas/sodium vapor mixture. A value of 1.4 is assumed in the PLUTO2 code.

K_{N1} = MNL = adiabatic compressibility of liquid sodium.

The fission-gas pressure P_{fi} and pressure P_{Nv} due to sodium vapor are obtained as explained in Section 14.4.5 using the following equations.

$$P_{\text{fi}} = \frac{R_{\text{fi}} \rho'_{\text{fi}} T_{\text{Na}}}{\theta_{\text{vg}}} \tag{14.4-240}$$

$$P_{\text{Nv}} = \frac{\text{RGNA} \cdot \rho'_{\text{Nv}} T_{\text{Na}}}{\theta_{\text{vg}}} \tag{14.4-241}$$

After evaluating the two upper limits $\Delta t_{\text{PL,pin}}$ and $\Delta t_{\text{PL,ch}}$ using Eq. (14.2-69) and Eq. (14.4-238), the PLUTO2 time step Δt_{PL} is taken to be the smaller of the two.

$$\Delta t_{\text{PL}} = \min\left[t_{\text{PL,pin}}, t_{\text{PL,ch}}\right] \tag{14.4-242}$$

If the value of Δt_{PL} obtained from Eq. (14.4-242) is less than the minimum time step given by the input parameter DTPLIN (suggested value 2.5×10^{-5} s), then Δt_{PL} is set equal to DIPLIN. Also, the PLUTO2 time step Δt_{PL} is not allowed to exceed a maximum value of 2×10^{-4} s (a number built in the code). The value of Δt_{PL} obtained in this way is rounded to an integral multiple of 1.0×10^{-5} s. The PLUTO2 time steps are not allowed to span the coolant dynamics time-step boundaries, or the heat-transfer time-step boundaries, or the primary loop time-step boundaries.

14.5 Temperature Calculation of Cladding, Structure, Reflector, and Liquid Sodium

14.5.1 Liquid Sodium, Cladding, Structure, and Reflector Temperature Calculation Outside of the Interaction Region

The expulsion of coolant from the core after PLUTO2 initiation (due to fuel-coolant interaction) results in significant preheating of cladding and other structures prior to the passage of the void interface, especially for the lower liquid slug, which has a large axial thermal gradient. Consequently, it is necessary to continue to compute the temperatures outside of the interaction zone after PLUTO2 initiation in order to provide accurate updated initial temperatures for cells being added t the integration zone (i.e., voided region). Core cladding temperatures outside of the interaction zone are computed in PLHTR, which is a modified version of the standard SAS [in heat-transfer model TSHTRV. Special routines were developed to compute the liquid coolant, structure, plenum cladding, and reflector temperatures outside of the interaction zone; a description of these subroutines follows.

Coolant temperatures are computed (in subroutine PLCOOL) based on a heat-transfer time step, as are those of the wetted structure, cladding, and reflector. The heat-transfer time step may be altered (in PLUDRV) to satisfy a Courant condition in either slug, based on the instantaneous sodium velocity.

The PLUTO coolant nodes, unlike those in the SAS boiling model, are centered in the numerical cell in order to be consistent with the PLUTO node structure in the interaction zone. The finite-difference equation, given below, is time explicit and uses donor cell differencing for the convective term:

$$A_{\text{ch}}\rho_{\text{N1}}C_{\text{p,N1}}L_{\text{i}}\left(T^{n+1}_{\text{N1,i}}-T^{n}_{\text{N1}}\right)+\Delta M_{\text{N1,i}}C_{\text{p,N1}}\left(T^{n}_{\text{N1,i}}-T^{n}_{\text{N1,i-1}}\right)=H^{l}_{\text{i}}L_{\text{i}}\Delta t_{\text{Ht}}\text{ , for } W_{\text{N1}}>0 \qquad (14.5\text{-}1)$$

where

$A_{\text{ch,i}}$ = local flow area,

L_{i} = wetted length in axial cell i,

$T^{n}_{\text{N1,i}}$; $T^{n+1}_{\text{N1,i}}$ = nodal coolant temperatures at times t and $t+\Delta t$.

$\Delta M_{\text{N1,i}}$ = mass transport into axial cell I during time interval Δt.

H^{l}_{i} = heat-transfer rate from cladding and structure per unit length.

ρ_{N1} = coolant density.

$C_{\text{p,N1}}$ = coolant specific heat.

Δt_{Ht} = heat-transfer time step.

W_{N1} = liquid sodium mass flowrate.

A similar equation is used for the case of downflow.

Normally L_{i} is set equal to the cell length:

$$L_{\text{i}}=z_{\text{i+1}}-z_{\text{i}} \qquad (14.5\text{-}2)$$

where z_{i} is the elevation of the lower cell boundary. Also, the mass transport ΔM_{i} is usually taken to be the product of the mean flowrate W_{N1}, computed from the interface displacement, times Δt_{Ht}. Exceptions are made for cells containing the void interfaces.

Consider the mesh cell containing the upper interface of the lower slug. For this case the wetted length is given by

$$L_{\text{i}}=z^{n+1}_{\text{if}}-z_{\text{i}} \qquad (14.5\text{-}3)$$

where z^{n+1}_{if} = interface elevation for the lower slug at the end of the heat-transfer time step and z_{i} = location of the fixed mesh cell boundary below the slug interface. For the case of expulsion (negative velocity in the lower slug), the inflow into the end cell is zero unless the interface crossed the upper boundary:

$$\Delta M_{\text{N1,i}}=\begin{cases}0 & \text{for } z^{n}_{\text{if}}<Z_{\text{i+1}}\\ -A_{\text{ch,i}}\left(1-f\right)\rho_{\text{N1,i}}\left(z^{n+1}_{\text{if}}-z_{\text{i+1}}\right) & \text{for } z^{n}_{\text{if}}\geq z_{\text{i+1}}\end{cases} \qquad (14.5\text{-}4)$$

where

f = input volume fraction CINAFO of liquid film left behind.

z^{n}_{if} = interface elevation for lower slug at time t.

For the case when the lower slug first reenters a cell, the coolant temperature for the end cell is set by $T^{n+1}_{\text{ch,i}}=T^{n+1}_{\text{ch,i-1}}$. In subsequent time steps, the end node temperature (for reentry or $W_{\text{N1}}>0$) is computed with the

reduced wetted length and with $\Delta M_{\mathrm{i}} = W_{\mathrm{N1}} \Delta t_{\mathrm{Ht}}$. Similar reasoning is applied to the treatment of the segment containing the lower interface of the upper slug.

The reflector, gas plenum cladding, and structure temperatures outside of the interaction zone are computed in subroutine FLSTR using straightforward, explicit-time-differenced equations. Like the coolant calculation, these temperatures are computed every heat-transfer time step. The difference equations are based on the same nodal structure as used in the pre-PLUTO2 SAS4A calculations and used the same FORTRAN variable names, which eliminates the need to initialize these variables when PLUTO2 calculations are begun. Reinitialization is required, however, during coolant reentry.

14.5.2 Cladding and Reflector Temperature Calculation in the Interaction Region

The transient cladding, reflector, and structure temperatures within the interaction region are computed in subroutine PLTECS, which is called form subroutine PLUDRV every PLUTO time step (see Figure 14.5.1). In the interaction region, both the cladding and reflector have three radial nodes (per axial segment), each of which has a different FORTRAN name. However, a temporary radial temperature array is defined with the numbering scheme shown in Figure 14.5.1 for the purpose of facilitating the solution of the radial heat conduction problem.

To evaluate cladding temperatures more accurately, one desires a constantly updated fuel surface temperature (every PLUTO2 time step) rather than the one computed from PLHTR every heat-transfer time step. This updated fuel surface temperature is obtained by extending the nodal structure to overlap that of the fuel model as shown in Figure 14.5.1. By overlapping, the fuel surface temperature is computed along with the cladding temperatures every PLUTO2 time step. To insure consistency with the fuel model, the temperature of node 2 (fuel surface) is reset along with that of node 1 every heat-transfer time step. The fuel temperature calculation in PLHTR, in turn, uses an integrated fuel-heat-loss boundary condition obtained from the cladding model.

The transient heat-transfer equation for each node is expressed in the following standard form:

$$(\mathrm{MC_p})_{\mathrm{i}} \frac{\mathrm{dT_i}}{\mathrm{dt}} = (ha)_{\mathrm{i\text{-}1}} (T_{\mathrm{i\text{-}1}} - T_{\mathrm{i}}) + (ha)_{\mathrm{i}} (T_{\mathrm{i+1}} - T_{\mathrm{i}}) + Q^l_{\mathrm{i}} \tag{14.5-5}$$

where

T_{i} is the nodal temperature.

$(\mathrm{MC_p})_{\mathrm{i}}$ is the heat capacity of the control volume per unit length.

$(ha)_{\mathrm{i}}$ is the coefficient of heat transfer from node I to $I+1$ per unit length,

Q^l_{i} is the control volume heat-generation rate per unit length.

The coefficients $(\mathrm{MC_p})_{\mathrm{i}}$ and $(ha)_{\mathrm{i}}$ are evaluated by the following relations:

$$(\mathrm{MC_p})_{\mathrm{i}} = \pi \left(r_{\mathrm{i}}^2 - \hat{r}_{\mathrm{i\text{-}1}}^2\right) \rho_{\mathrm{i+1}} C_{\mathrm{p,i\text{-}1}} + \pi \left(\hat{r}_{\mathrm{i}}^2 - r_{\mathrm{i}}^2\right) \rho_{\mathrm{i}} C_{\mathrm{p,i}} \tag{14.5-6}$$

$$(ha)_{\mathrm{i}} = \frac{2\pi r_{\mathrm{i}} \hat{k}_{\mathrm{i}}}{(r_{\mathrm{i+1}} - r_{\mathrm{i}})}, \text{ for } i \neq 2, 5 \tag{14.5-7}$$

$$(ha)_{\mathrm{i}} = 2\pi r_{\mathrm{i}} \hat{h}_{\mathrm{i}}, \text{ for } i = 2, 5 \tag{14.5-8}$$

where ρ_{i}, $C_{\mathrm{p,i}}$, and k_{i} are the density, specific heat, and thermal conductivity of the material between nodes i and $i+1$. Based on the numbering scheme shown in Figure 14.5.1, the quantities k_3 and k_4 are set equal to the input value of the cladding thermal conductivity DCL. The products $\rho_3 C_{\mathrm{p,4}}$ and $\rho_4 C_{\mathrm{p,4}}$ are set equal to the input parameter CPCLRH. The mean radius $\hat{r}_{\mathrm{i}}$ is defined by

$$\hat{r}_{\mathrm{i}} = 1/2 \, (r_{\mathrm{i}} + r_{\mathrm{i+1}}) \tag{14.5-9}$$

where r_i is the radius of node i.

Cladding melting is accounted for by using an augmented clad heat capacity, C'_p, when the clad temperature falls in the melting band, where C'_p is defined by:

$$C'_p = C_p + \frac{\lambda}{\Delta T_{me}} \tag{14.5-10}$$

where

C_p = is the normal specific heat of the solid cladding given by the input parameter CPCL.

λ = is the latent heat of fusion which is evaluated as the difference between the input values of energies of cladding at liquidus and solidus, EGSELQ and EGSESO.

ΔT_{mc} = is the difference in the liquidus and solidus temperatures of the cladding which are input (see TESELQ and TESESO, respectively).

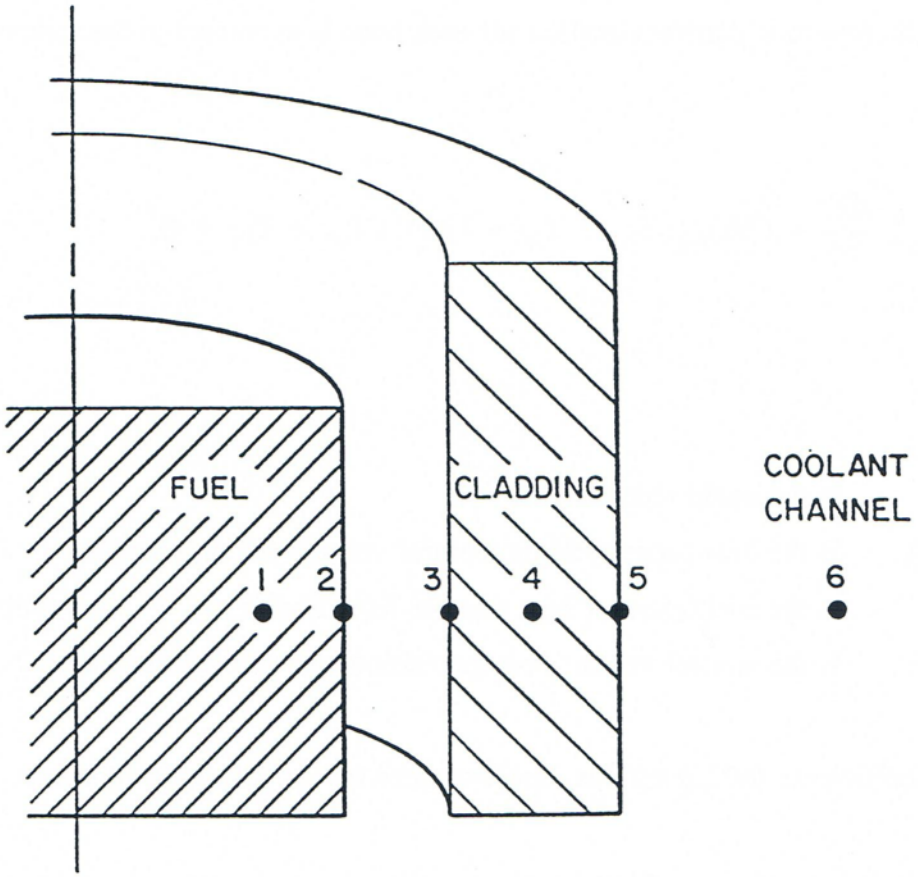


Figure 14.5.1: Radial Node Numbering Scheme Used in Subroutine PLTECS for the Temperature Calculation of Cladding and Reflector in the Interaction Region

When entering or leaving the melting band, an adjustment o the computed temperature is required to insure that energy is properly conserved in the system. The specific heat of the cladding above the liquidus temperature is given by the input value CPSE.

The coefficient of heat transfer h_2 is set equal to the gap coefficient in the active fuel and blanket regions. In the plenum and reflector regions, however, h_2 is set equal to zero to simulate an adiabatic boundary condition at

the inner cladding surface. The form of the equations for $(\mathrm{MC_p})_i$ and $(ha)_i$, for the reflector region is slightly modified to correspond to a slab geometry, rather than a cylindrical one.

The heat-transfer coefficient (ha), and effective sink temperature T_6 for the outer surface of the cladding and reflector are given by:

$$(ha)_5 = (ha)_{\text{Na,cl}} + (ha)_{\text{fu,cl}} + (ha)_{\text{ff,cl}} \tag{14.5-11}$$

$$(ha)_5 = (ha)_{\text{Na,cl}} + (ha)_{\text{fu,cl}} + (ha)_{\text{ff,cl}} \tag{14.5-12}$$

The three terms on the right-hand side of these equations account for heat transfer from the cladding (or structure) to each component in the channel: namely, sodium (and fission gases), moving fuel, and stationary frozen fuel.

Eq. (14.5-12) is converted to a fully implicit difference equation in time. This scheme was used to obtain numerical stability when analyzing thin cladding without using excessively small PLUTO time increments. This feature will be especially useful in the analysis of ablating clad where the nodal heat capacities are shrinking to zero. The resulting difference equations for each axial slice of the core are a set of simultaneous equations for the nodal temperatures, which are solved by the Thomas algorithm for tri-diagonal coefficient matrices.

14.5.3 Structure Temperature Calculation in the Interaction Region

The temperature calculation in the subassembly hexcan wall or in the flow tube of an experiment test section normally uses the same two-node mesh structure as the remainder of the SAS4A code. However, when the SAS4A input specifies one large and one rather small node width ($w_{\text{small}} < 0.1 w_{\text{sr}}$), the width of the small node is set to $w_{\text{small}} = 0.1 w_{\text{sr}}$. This is done in order to avoid stability problems in the explicit temperature calculation performed in PLUTO2 or in LEVITATE. As for the pre-failure calculation of the SAS node, it is recommended that the structure node facing the coolant channel should be considerable thinner than that for the structure node facing the intersubassembly gap. This is because the node facing the coolant channel should be capable of rapidly responding to changes in the transfer from coolant or molten fuel.

In order to treat the heat transfer between two unequal nodes accurately, an approach is used in which the temperature profile in the structure is approximated by a parabola rather than a straight line between the two temperature nodes. This approach, which was originally developed for LEVITATE, has been adopted in PLUTO2. It is described in detail in Section 16.5.7.2.

14.6 Interaction with the Point Kinetics and the Primary Loop Module

14.6.1 Interaction With the Point Kinetics Module

In Section 14.1.2, the calculation in PLUTO2 of the specific power in the fuel and the total power in an original pin node was described. This was also discussed later when the heat source terms in the fuel pin and in the channel were described. The calculation of the reactivity feedbacks for channels in which PLUTO2 is active is based on fuel temperature and mass distributions and coolant voiding distributions calculated in PLUTO2 and passed to subroutine FEEDBK, where the reactivity feedbacks are calculated.

In PLUTO2, the fuel mass in the fuel pin nodes can be relatively easily calculated because the fixed Eulerian grid used for the in-pin fuel motion is part of the grid on which the material worths are defined (see Figure 14.1.4). The calculational grid in the coolant channel has additional cells above and below the pin grid and is therefore indexed differently. However, the axial spacing of the channel grid corresponds to that of the fuel-pin grid. A problem associated with the calculation of the fuel or sodium masses on the channel grid is the existence of

partial Lagrangian cells at the edges of the interaction region. Partial Lagrangian cells such as the channel cell IFMIBT in Figure 14.1.4 contain fuel and sodium that extend into the adjacent cell IFMIBT-1. For the purpose of the reactivity calculation this fuel and sodium are included in cell IFMIBT-1 as opposed to the approach in the hydrodynamics.

14.6.2 Coupling with the Primary Loop Module

The coupling between PLUTO2 and the primary loop module is quite simple when the PRIMAR-1 option has been chosen (i.e., input parameter IPRION set to a value not equal to 4). In this case, PLUTO2 uses the constant outlet coolant plenum pressure PX which is input and an inlet coolant plenum pressure which is determined by PRIMAR-1. PRIMAR-1 calculates the steady-state pump head that is multiplied by an input table or function during the transient (see Section 5.9). Moreover, PLUTO2 also uses the table input for the inlet temperature history and the single input value TUPL for the reentry temperature at the outlet. However, PLUTO2 will not feed back any information to the primary loop module if the PRIMAR-1 option has been chosen.

If the PRIMAR-4 option has been chosen (IPRION=4), PLUTO2 will use the time-dependent inlet and outlet pressures which are calculated by PRIMAR-4. This is done in the following way:

$$P_{\text{inlet}}(t) = P_{\text{inlet}}(t_{\text{PR1}}) + (t - t_{\text{PR1}}) \cdot \frac{\partial \text{P}_{\text{inlet}}}{\partial \text{t}} \tag{14.6-1}$$

where

$P_{\text{inlet}}(t_{\text{PR1}})$ is the PRIMAR-4 calculated inlet pressure at the beginning of the current primary loop time step.

$\frac{\partial \text{P}_{\text{inlet}}}{\partial \text{t}}$ is the PRIMAR-4 calculated rate or inlet pressure change during the current primary loop time step.

The outlet pressure is calculated in the same manner:

$$P_{\text{outlet}}(t) = P_{\text{outlet}}(t_{\text{PR1}}) + (t - t_{\text{PR1}}) \cdot \frac{\partial \text{P}_{\text{outlet}}}{\partial \text{t}} \tag{14.6-2}$$

PLUTO2 also uses the time-dependent inlet and outlet temperatures calculated by PRIMAR4. Since inlet and outlet temperatures change slowly, only the average values over each PRIMAR-4 step are used in PLUTO2.

When the PRIMAR-4 option has been chosen, PLUTO2 provides PRIMAR-4 with total sodium masses ejected into or received from the inlet or outlet plena during a primary loop time step:

$$\Delta M_{\text{Na,ic,inlet}} = N_{\text{subas,ic}} \int_{t_{\text{PR1}}}^{t_{\text{PR2}}} W_{\text{Na,inlet}} \text{dt} \tag{14.6-3}$$

and

$$\Delta M_{\text{Na,ic,outlet}} = N_{\text{subas,ic}} \int_{t_{\text{PR1}}}^{t_{\text{PR2}}} W_{\text{Na,outlet}} \text{dt} \tag{14.6-4}$$

where

ic = SAS4A channel number.

t_{PR1} = time at the beginning of the PRIMAR-4 time step.

t_{PR2} = time at the end of the PRIMAR-4 time step.

W_{Na} = sodium liquid and/or vapor mass flow rate.

PLUTO2 also provides PRIMAR-4 with the channel mass flow rates at the end of the primary loop time step. As long as pure liquid sodium is ejected into or received from the upper and lower plena, temporal integrals over the sodium mass flow rate times the temperature of the ejected sodium are also provided by PLUTO2.

However, when the upper liquid sodium slug has been ejected out of the subassembly outlet, the additional heat added to the outlet plenum by the subsequently ejected two-phase sodium (which condenses in the plenum) and the ejected fuel during a primary loop time step is calculated by PLUTO2 for use in PRIMAR-4 (see Section 5.11.1):

$$\Delta E_{\text{v,ic}} = [\lambda_{\text{Na}} \cdot x_{\text{Na}} \cdot W_{\text{Na}} + u_{\text{fu}} \rho'_{\text{fu}} \cdot \text{AXMX} \cdot \{e_{\text{fu}} - \text{EGFUTE}\,(T_{\text{Na}})\}] \\ N_{\text{sibas,ic}} \cdot (t_{\text{PR2}} - t_{\text{PR1}}) \tag{14.6-5}$$

where

$x_{\text{Na}}, \lambda_{\text{Na}}$ is the quality of the two-phase sodium and the enthalpy of evaporation of sodium in the highest coolant node is SAS4A channel ic.

EGFUTE See Eq. (14.4-179).

The quantities $\rho'_{\text{fu}}, e_{\text{fu}}, T_{\text{Na}}$ in Eq. (14.6-5) all refer to the highest coolant node HTP. A similar equation is used for the inlet plenum also. Eq. (5.11-2) in Chapter 5, which calculates an estimate of the flow into or out of each SAS4A channel, requires several coefficients for each SAS4 channel. These coefficients are calculated before boiling by the single-phase hydraulics module, then by the boiling module, and, after fuel-pin failure, they are calculated by PLUTO2 or LEVITATE, depending on which module is active in a certain channel.

14.7 Code Logic Description

14.7.1 PLUTO2 Initialization

For the initialization of PLUTO2, the flow diagrams in Figure 14.7.1 and 14.7-2 are relevant. The flowchart in Figure 14.7.1 shows part of the logic flow in subroutine FUINIT (**FU**EL MOTION **INI**TIALIZATION) in which the decision is made whether PLUTO2 or LEVITATE should be initialized once subroutine FAILUR (which is called from the DEFORM Transient driver DFORM3 or from TSTHRM if cladding motion has already begun) has predicted pin failure. A minimum fuel melt fraction equal to the input parameter FMELTM must exist in the failing node before PLUTO2 or LEVITATE is allowed to be initiated. The decision which one of the two modules, PLUTO2 or LEVITATE, is to be initiated depends on the existence and size of a boiling region at the time of failure and whether the pins are predicted to fail into a voided or unvoided region of the channel. If there is no boiling region, as in a TOP accident, or if the pins fail into the liquid region of a partially voided channel, PLUTO2 will be initiated because of its capability of treating FCIs and sweepout of particulate fuel. If there is a large voided region in the channel, in which cladding motion may already have started, LEVITATE, which is designed for voided-channel fuel motion analysis, will be initiated. For the case of fuel pins failing into a multi-bubble boiling region, PLUTO2 will be initiated if the average void fraction in this region is below 70% at the time of pin failure. If the average void fraction is greater than 70%, it will be a LEVITATE case which has to be started by first initiating PLUTO2. In this case, the flag ILEPLI is set to 1. The flow diagram in Figure 14.7.2 shows that a switch to LEVITATE will be immediately made in this case, once the PLUTO2 driver routine (PLUDRV) is entered.

The flow diagram shown in Figure 14.7.2 shows how PLUTO2 and LEVITATE are actually initiated. In a PLUTO2 case, a check is first made to determine whether cladding motion has already begun. This is an unlikely situation that cold only come about if liquid sodium reentered into the boiling region after cladding motion had begun. This would have prevented the selection of LEVITATE. Since PLUTO2 cannot handle this situation, control is returned to subroutine FAILUR, and no fuel motion initiation will take place until the run has proceeded further and led to a more extensive boiling region that allows the initiation of LEVITATE.

In subroutine PLSAIN (**PL**UTO2 **S**AS4A COOLANT CHANNEL **IN**ITIALIZATION), the coolant mass, temperature and velocity distributions are initialized. In the case of a nonboiling channel, this is relatively straightforward, whereas it is quite complex to properly translate these quantities from the Lagrangian grid of a multi-bubble boiling module to the fixed Eulerian PLUTO2 grid.

In subroutine PLINPT (**PL**UTO2 **IN**PU**T**), the coolant channel geometry is set up, all arrays relevant for the multi-component flow in the channels are initialized, and the geometry of the molten fuel cavity in the pins is established. The size of the latter depends on the radial and axial extent of fuel melting calculated by TSHTRN (for nonboiling cases) or TSHTRV (for cases in which boiling had already begun) and on the fuel-pin grid deformation calculated by DEFORM. The radially averaged internal fuel energy in the molten cavity is calculated based on the radial temperature profile in the cavity. The total fuel and dissolved fission-gas masses in all cavity nodes are also calculated. The latter are based on the DEFORM calculated values.

In subroutine PLSET (**PL**UTO2 **SET**UP), the molten cavity pressure, which is calculated by DEF"ORM and which can also be input (see input value PCFAIL) for parametric studies, is used to determine the mass of free fission gas in each cavity node. This was discussed earlier in more detail in Section 14.2.2. All other arrays necessary for the in-pin calculation are also initialized. Moreover, several channel arrays for the cladding and *structure* temperature calculations in the interaction region are initialized in this subroutine.

Subroutine PLSET1 (**PL**UTO2 **SET**UP**1**) initializes parameters necessary for the temperature calculation in the liquid coolant slugs, which is performed in subroutine PLCOOL. Moreover, parameters necessary for the temperature calculation in the structure, plenum cladding and reflectors outside the PLUTO2 interaction zone (which is performed in subroutine PLSTR) are initialized in PLSET1.

The flowchart in Figure 14.7.2 also shows that LEVITAE uses most or all of the PLUTO2 initialization routines. However, LEVITATE also needs subroutine LESAIN for the case of a single large initial boiling bubble and subroutine LECLIN if cladding motion has started prior to pin failure. It can also be seen in Figure 14.7.2 that the flag ICALC is set to 2 on the LEVITATE branch and 3 on the PLUTO2 branch. This flag controls whether the LEVITATE driver routine LEVDRV or the PLUTO2 driver routine PLUDRV will be called when the SAS4A transient driver TSTHRM is executed during the next coolant time step.

14.7.2 Time Step Considerations and Auxiliary Subroutines

The PLUTO2 flow logic has already been described in considerable detail in Section 14.1.2. Here, only the different time steps involved and the calling of auxiliary subroutines will be discussed.

The flow diagram of the PLUTO2 driver routine PLUDRV, which already appeared in Section 14.1.2, is also shown in Figure 14.7.3 in order to have a complete set of PLUTO2 flow diagrams in this section.

The PLUTO2 driver routine can be called by the SAS4A transient driver TSTHRM at the beginning or at any time during a primary loop time step. The latter is common for all SAS4A calculational channels and is smaller than or equal to the main (point kinetics) time step, which is also common for all SAS4A channels.

Before PLUDRV is called for the first time in a given channel, the PLUTO2 time (TIMEPL), which is counted from the time of pin failure, is set to zero in the FAILUR routine. Moreover, the time of pin failure relative to the time of the accident initiation is recorded in the channel-dependent variable TMPLIN.

When PLUDRV is entered, TIMEPL is advanced by adding the PLUTO2 time step right after subroutine PLSET2 (which initializes temporary variables) has been called (see fig. 14.7-3). The very first PLUTO2 time-step size used is the input constant DTPLIN which is the initial and the minimum PLUTO2 time-step size. After the subroutines PLIPIN and PL2PIN have performed the in-pin calculation, a new time-step size for the in-pin motion is calculated as described in Section 14.2.8. Once all the subroutines calculating the channel fuel and sodium/gas dynamics have been called, a new time-step size for the in-pin motion is calculated as described in Section 14.2.8. Once all the subroutines calculating the channel fuel and sodium/gas dynamics have been called, a new time-step is calculated for the channel hydrodynamics (see Section 14.4.6.4). This is compared with the time-step size calculated for the in-pin motion and the smaller of the two will be used as the new PLUTO2 time-step size if it is larger than the above-mentioned input value DTPLIN. If it is smaller than DTPLIN, the latter will be used as the new PLUTO2 time-step size. Also, if it is larger than 2×10^{-4} s, it will be set to this value. Moreover, if the newly determined time step extends beyond the end of a heat-transfer time step, it will be cut back such that the new PLUTO2 time will coincide with the end of the heat-transfer time step.

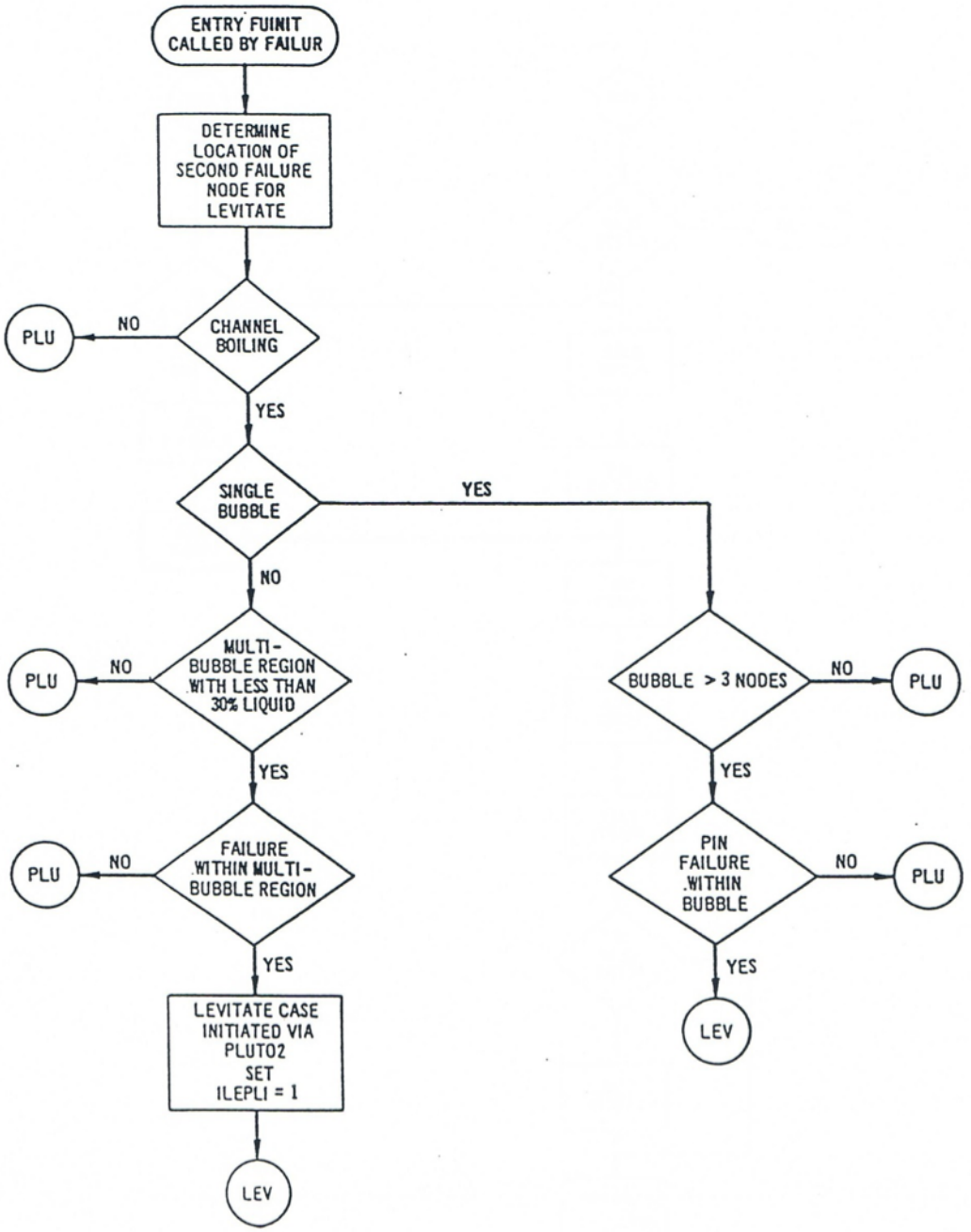


Figure 14.7.1: First Part of the Flow Chart of Subroutine FUINIT Showing the Logic Used in Determining Whether the PLUTO2 Module of the LEVITATE Module is to be Activated. The Branch PLU or LEV Implies that the PLUTO2 Module of the LEVITATE Module is Respectively Activated. The Details of Branches PLU and LEV are Shown in Figure 14.7.2.

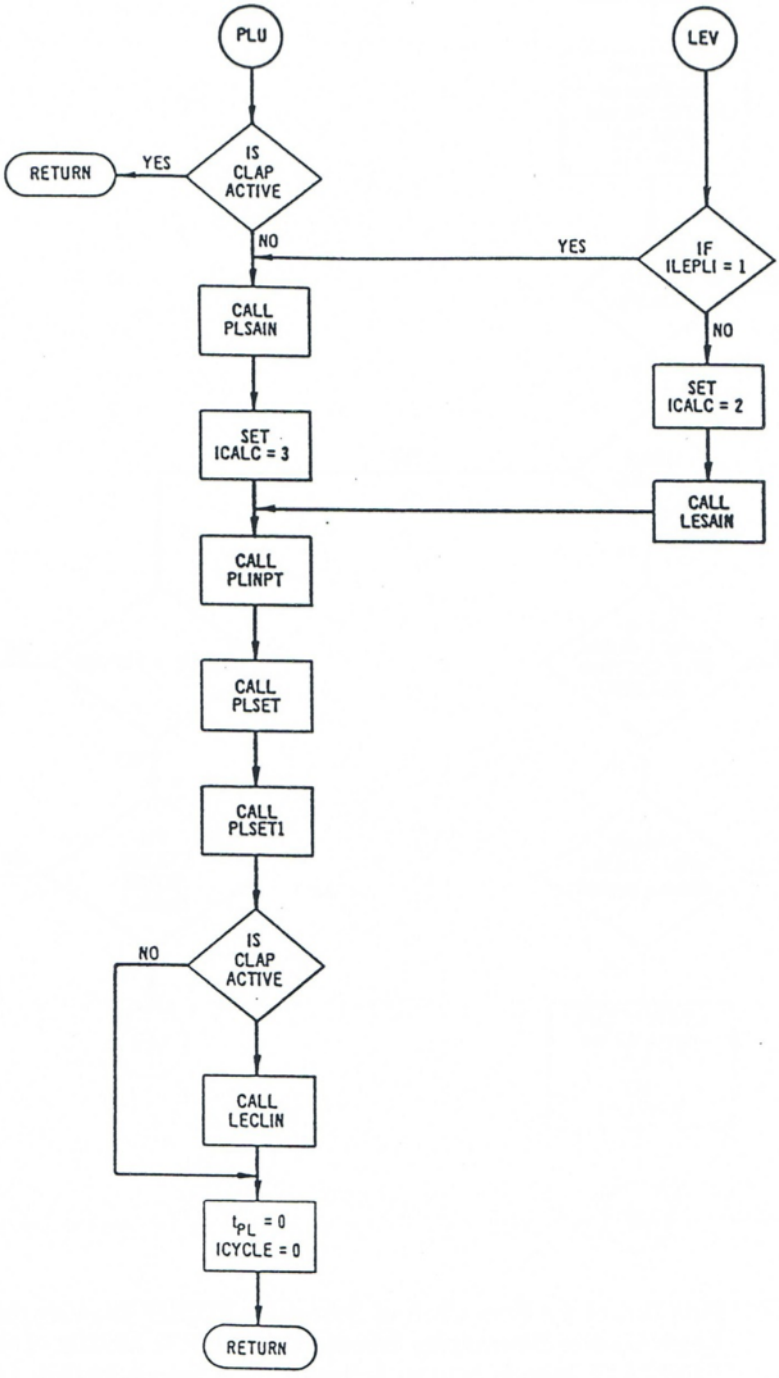


Figure 14.7.2: Second Part of the Flow Chart of Subroutine FUINIT Showing the Initialization of the PLUTO2 and LEVITATE Modules

This is possible since the heat-transfer subroutine PLHTR calculates only the fuel temperatures in the region determined by the interaction region (see Figure 14.1.4 in Section 14.1) and the fuel and cladding temperatures outside the interaction region. None of these temperatures are changing rapidly relative to PLUTO2 time-step sizes.

The determination of the heat-transfer time step is made using a series of tests. First, the initial and maximum heat-transfer time step is set to 1 ms, which is about an order of magnitude smaller than the characteristic heat-transfer time for cladding. If the initial or any later time step is larger than the current primary loop time step, it is set equal to the latter. Since the primary loop time step has to be less than or equal to the main (point kinetics) time step, it is, of course, also limited by the main time step.

The heat-transfer time step is not only used for the heat-transfer calculation of all the fuel and the cladding reflector, and the structure temperature calculation outside the interaction region, but also for the temperature calculation in the liquid sodium slugs. Since the latter involves not only heat conduction from the walls but also heat convection from the coolant cells above and below the one considered, the heat-transfer time step is also limited by a Courant condition based on the liquid slug velocity.

The flow diagram in Figure 14.7.3 shows that PLUTO2 retains control and advances the solution using PLUTO2 time steps until the end of the primary loop time step is reached. This is different from the single-phase hydraulics driver TSCL0 and the boiling module driver TSBOIL, which return control to the SAS4A transient driver TSTHRM after every coolant time step. The latter is the basic time step in the modules controlled by TSCL0 and TSBOIL. In PLUTO2 (and LEVITATE) the coolant time step is set equal to the primary loop time step. PLUTO2 or LEVITATE thus also return control to the transient SAS4A driver at the end of this "artificial" coolant time step.

Several auxiliary subroutines used in the PLUTO2 module are not shown in the flow chart in Figure 14.7.3. These include the function subroutine TEFUES, which calculates fuel temperatures for a given internal fuel energy and TESEEG which calculates steel temperatures for a given internal steel energy. Function subroutines EGFUTE and EGSETE perform the inverse operation for fuel and steel, respectively, i.e., calculating internal energies for a given temperature.

Several auxiliary subroutines and subroutine functions are called by the subroutine PLHTR performing the heat-transfer calculation in the solid annulus of the fuel pin and in the cladding outside the interaction region. PLHTR is a derivative of subroutine TSHTN3 described in more detail in Section 3.3.1 of Chapter 3. The most important auxiliary subroutine called by PLHTR is INVRT3, which inverts a tri-diagonal matrix. Subroutines KFUEL and KCLAD calculate the fuel and cladding conductivity, respectively. Subroutine CFUEL calculates the heat capacity of the fuel and TSHTN5 corrects fuel and cladding temperatures for the heat of fusion, if melting is occurring. Subroutine HBSMPL and function subroutine HBFND calculate the gap conductance.

14.8 Description of Input to and Output of the PLUTO2 Module

14.8.1 PLUTO2 Input

The SAS4A input variables required specifically by the PLUTO2 fuel motion model are given in Table 14.8.1 with their suggested values in MKS units (kilogram, meter, second, degree Kelvin). Also indicated in Table 14.8.1 are some particular sections of the documentation in which the input variables are explained, or some particular equations in which the variables are used. The first column of the table gives the location of the variable in the complete input deck of the SAS4A code. The second and the third columns give the Fortran variable names used in the code and the SAS4A input description, and the corresponding symbols, if any, used in the mathematical equations in the documentation.

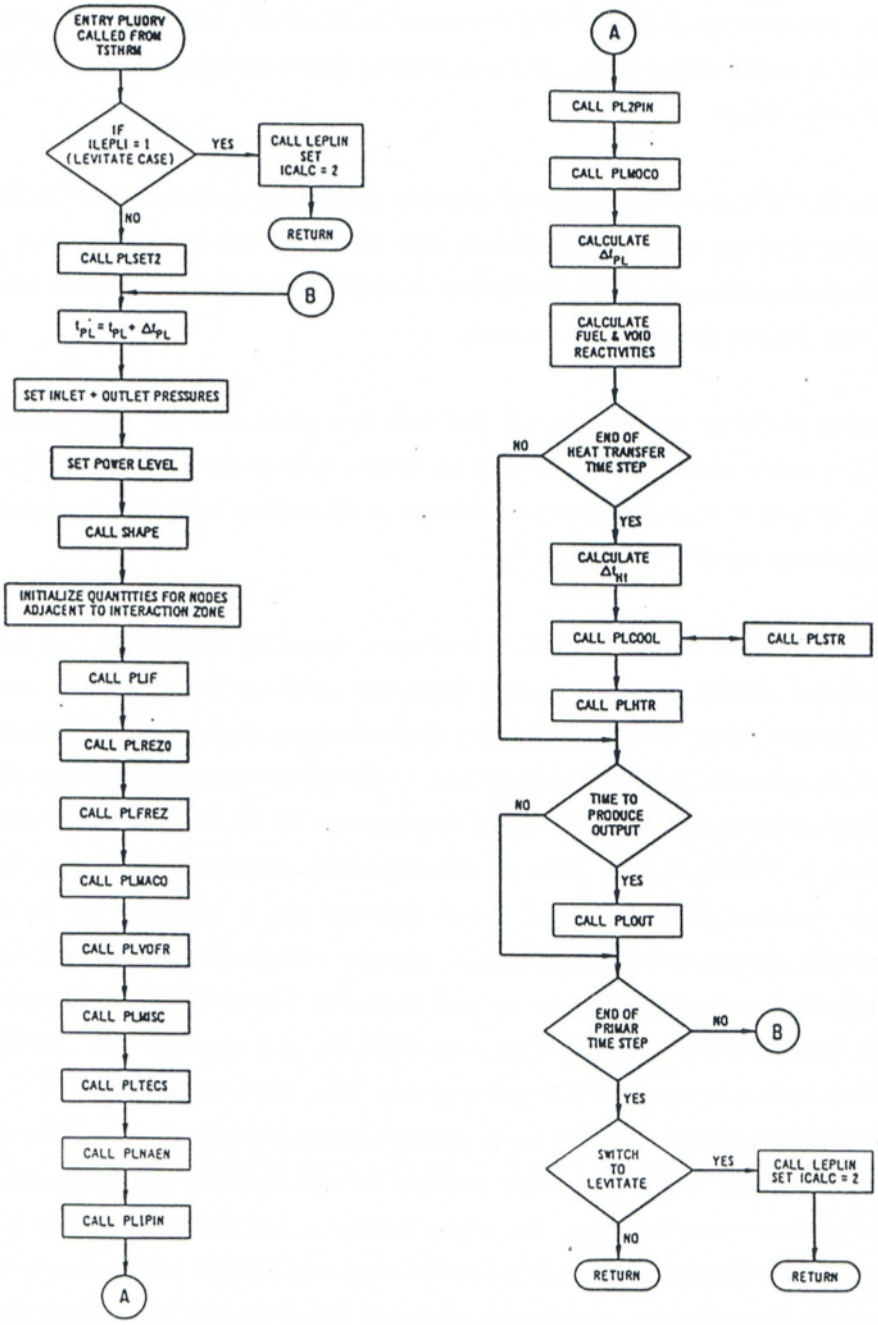


Figure 14.7.3: Flow Chart of the PLUTO2 Driver Sub routine PLUDRV

14.8.2 PLUTO2 Output

The main PLUTO2 output (see Figure 14.8.1) for a given SAS4A channel (the channel number is shown at the top of each output page) is separated into two major sections. One of these sections (output items 1 to 7 in Figure 14.8.1) deals with the in-pin motion and the fuel and fission-gas ejection from the pins and the other section (output items 8 to 18 in Figure 14.8.1) deals with the motion of fuel, sodium, and fission gas in the coolant channels. *All output is strictly in SI units* (e.g., kg, m, s, K, joules, pascals, etc.). At the time of entry to PLUTO2 module, the variables of output item 18 are printed with the message "PLUTO STARTS" and then follow output items 1 to 18.

Table 14.8.1: List of Input Variables Required by the PLUTO2 Fuel Motion Model

SAS4A Input Location	Fortran Variable	Symbol	Describing Section	Equation Number	Suggested Value in MKS Units	Comments
Block 1, INPCOM						
37	KFAILP		14.3.3		0	Mechanistic axial propagation of pin failure is opted.
38	NCPLEV		14.3.4, *14.4.3.2*		3	The fuel motion calculation is switched from PLUTO2 to LEVITATE when 3 axial cladding nodes have exceeded the cladding liquidus temperature.
Block 13, PMATCM						
1124	RGAS		14.2.6	Eq. (14.2-40)	130.637	
1125	CINAFO			Eq. (14.4-43), Eq. (14.4-44), Eq. (14.4-96)		For a consistent input, the coolant film volume fraction (CINAFO) should be evaluated from cladding and structure film thicknesses WFO and WFSOO of channel 1. If CINAFO so evaluated for another channel disagrees with the above value, a consistent input is not possible.
1126	CIBBIN		14.4.3.1		0.7	See Section 14.4.3.1.
1127	CIREFU			Eq. (14.4-205)	2100.0	

continues on next page

Table 14.8.1 – continued from previous page

SAS4A Input Location	Fortran Variable	Symbol	Describing Section	Equation Number	Suggested Value in MKS Units	Comments
1128	CIFRFU			Eq. (14.4-205)	0.02	Page 264 of Ref. [14-49].
1129	CIFUMO			Eq. (14.4-205)	0.5	All intra-pin axial momentum of the fuel perhaps will not be retained after ejection into coolant channel.
1130	CIVOID		14.4.3.4.1.1		0.5	See Section 14.4.3.4.1.1.
1131	CIA1		14.4.3.4.2.1	Eq. (14.4-112)	1.0	A recent theoretical evaluation [14-51] gives CIA1 = 3.3 to 5.0 but a preliminary calibration of the PLUTO2 module in the SAS4A version 1.0 has been done with CIA1 = 1.0
1132	CIA2		14.4.3.4.2.1	Eq. (14.4-72), Eq. (14.4-116)	2.0	See Section 14.4.3.4.2.1.
1133	CIA3			Eq. (14.4-30), Eq. (14.4-121)	0.0158	This value is good for mixed oxide fuel but not for metal alloy fuel
1134	CIA4		14.4.3.4.1		0.6	CIA4 should e larger than CIVOID.
1135	CIA5			Eq. (14.4-195)	-1.7	Ref. [14.49, 14-63].
1136	CIA6		14.4.6.1.1	Eq. (14.4-202)	0.4272	
1139	CPFU	$C_{p,fu}$		Eq. (14.2-55), Eq. (14.4-166), Eq. (14.4-170)	500.0	Ref. [14-59].
1140	CDFU	k_{fu}		Eq. (14.4-152)	3.0	Ref [14-60].

continues on next page

Table 14.8.1 – continued from previous page

SAS4A Input Location	Fortran Variable	Symbol	Describing Section	Equation Number	Suggested Value in MKS Units	Comments
1141	CMNL	K_{Nf}		Eq. (14.4-50)	4.6D-10	The compressibility of liquid sodium is very temperature-dependent and this value [14-60] corresponds to 1200 K, roughly the sodium temperature in the vicinity of pin failure at the time of failure.
1142	CDNL	K_{Nl}		Eq. (14.4-89)	50.0	This is liquid sodium thermal conductivity at about 1200 K [14-59].
1143	CIETFU			Eq. (14.4-39), Eq. (14.4-41)	0.1	
1144	CDVG	k_{vg}		Eq. (14.4-104)	0.067	The thermal conductivity of a gas mixture is very dependent on its temperature and composition. This value of CDVG is sodium vapor thermal conductivity at 1500 K [14-61], roughly 300 K higher than the assumed sodium temperature in the vicinity of pin failure at the time of failure.
1145	VIFI				9.5D-5	This is xenon viscosity at about 1500 K extrapolated from temperature-dependent measured data [14-62] by fitting an equation of the form $T^{0.89}$ (T = xenon temperature). It is not yet used in the code.
1146	CFNACN			Eq. (14.4-99)	6.0D4	
1147	CFNAEV			Eq. (14.4-100)	6.0D5	

continues on next page

Table 14.8.1 – continued from previous page

SAS4A Input Location	Fortran Variable	Symbol	Describing Section	Equation Number	Suggested Value in MKS Units	Comments
1148	FIFNGB			Eq. (14.2-18)	0.1	Also see suggested values of DEF"ORM input in Chapter 8.
1149	VINL	μ_{Nl}		Eq. (14.4-95)	1.5D-4	This is liquid sodium viscosity at about 1200 K [14-61].
1150	VIVG	μ_{vg}		Eq. (14.4-95) Eq. (14.4-114)	2.2D-5	This viscosity of a gas mixture is very dependent on its temperature and composition. This value of VIVG is sodium vapor viscosity at about 1500 K [14-61].
1151	EGFUSO	$e_{fu,sol}$		Eq. (14.4-46), Eq. (14.4-55)	1.0D6	Ref. [14-59].
1152	DZPLIN		14.4.2.3		0.02	DXPLIN must be smaller than the shortest axial mesh cell in channel.
1153	CFCOFV			Eq. (14.4-117), Eq. (14.4-142)	6.0D4	This parameter is not yet used in the code.
1155	C1VIPR		14.2.4	Eq. (14.2-46)	3.0D-3	
1156	C2VIPR		14.2.4	Eq. (14.2-45)	0.0	
1157	SUFU	σ_{fu}		Eq. (14.4-201)	0.45	For a consistent input, this value should equal the value input in GAMMA [14-59].
1158	RAFPLA		14.4.3.1		2.5D-4	
1159	RAFPSM		14.4.3.1		2.5D-4	
1160	VFNALQ		14.4.3.1		0.33	See Section 14.4.3.1.

continues on next page

Table 14.8.1 – continued from previous page

SAS4A Input Location	Fortran Variable	Symbol	Describing Section	Equation Number	Suggested Value in MKS Units	Comments
1161	EGBBLY		14.4.3.2	Eq. (14.4-51), Eq. (14.4-55)	EGFUSO + UFMELT(1 × 0.5	Fuel freezing is initiated when the fuel internal energy drops below the energy corresponding to 0.5 melt fraction. The value of the latter energy should be evaluated by adding 0.5 of the latent heat of fusion of fuel type 1 (**UFMELT**) to the solidus internal energy.
1162	VIFULQ	$\mu_{\text{fu,liq}}$		Eq. (14.2-36), Eq. (14.2-47), Eq. (14.4-205)	4.3D-3	Ref. [14-59].
1163	VFNARE		14.4.3.1		0.8	VFNARE > VFNALQ
1164	DTPLIN	Δt_{PL}	14.2.8, *14.4.6.4*	Eq. (14.4-41), Eq. (14.4-29)	2.5D-5	
1165	AXMX		14.2.5, *14.4.1*	Eq. (14.2-2), Eq. (14.4-1)	1.0	Although the results of the code (except the volume fractions) are independent of the value of AXMS, the volume fractions are meaningful if AXMX is set equal to the cross-sectional if AXMS is set equal to the cross-sectional area within the outer surface of the subassembly hexcan.
1166	EPCH		14.2.4	Eq. (14.2-1)	1.0	
1167	TIPLMX		14.1.2 (last paragraph)		1.5	
1168	DTPLP				2.0D-3	

continues on next page

Table 14.8.1 – continued from previous page

SAS4A Input Location	Fortran Variable	Symbol	Describing Section	Equation Number	Suggested Value in MKS Units	Comments
1169	FNMELT		14.2.2, *14.2.3*	Eq. (14.2-12), Eq. (14.2-14)	0.9	
1170	CIRTFS			Eq. (14.2-21), Eq. (14.4-20)	16.67	
1172	CIFUFZ		14.4.3.3	Eq. (14.4-85)	1.0	It is not yet used in the code.
1173	TIFP		14.4.3.1		0.015	It makes no difference if both fuel particle radii are equal.
1174	CIANIN		14.4.3.3	Eq. (14.4-78)	0.5	
1175	TEFAIL			Eq. (14.4-78)	TESOL(1)	It should be set equal to the solidus temperature of cladding type 1 (location 810).
1176	FNARME			Eq. (14.3-10), Eq. (14.3-12)	0.9	
1177	PRFAIL			Eq. (14.3-13)	0.0	
1178	EGMN		14.4.3.1		EGFUSO + UFMELT(1 × 0.3	EGMN should be smaller than EGBBLY. It is assumed that continuous fuel flow regimes cannot be initiated below the energy corresponding to 0.3 melt fraction.
1179	HCFFMI			Eq. (14.4-123), Eq. (14.4-130)	1.0D5	
1180	HCFUBB			Eq. (14.4-137), Eq. (14.4-141)	3.0D4	
1181	FNHTFU				0.1	It is not yet used in the code.

continues on next page

Table 14.8.1 – continued from previous page

SAS4A Input Location	Fortran Variable	Symbol	Describing Section	Equation Number	Suggested Value in MKS Units	Comments
1184	TECLMN		14.4.3.2	Eq. (14.4-53)	TESOL(1)	The value should be chosen equal to the solidus temperature of cladding type 1 (**TESOL**) based on the consideration that cladding at solidus cannot provide support for freezing fuel.
1185	TECLRL		14.4.3.2	Eq. (14.4-65)	TELIQ(1)	See Section 14.4.3.2.
1186	CIHCFU		14.2.6	Eq. (14.2-35)	0.0158	See location 1133.
1187	HCCLMI			Eq. (14.4-102), Eq. (14.4-103)	1.0D5	
1188	CMFU	$K_{\mathrm{fu,liq}}$		Eq. (14.2-40)	6.0D-11	The compressibility of liquid mixed oxide fuel is very dependent on temperature and the chosen value is for fuel at 4000 K [14-60].
1195	CDCL		14.5.2	Eq. (14.5-7)	32.0	This value is solid cladding thermal conductivity averaged over the temperature range 1200 K to the cladding solidus temperature of 1700 K [14-59].
1196	CPCL	C_{p}	14.5.2	Eq. (14.5-10)	655.0	This value is the cladding specific heat averaged over the temperature range 1200 K to the solidus temperature of 1700 K [14-59].
1197	CPCLRH	$\rho_{\mathrm{i}} C_{\mathrm{p,i}}$	14.5.2	Eq. (14.5-6)	7400*CPCL	This value is obtained using the cladding specific heat CPCL (location 1196) and a cladding density of 74000 kg/m^3 averaged over the range 1200 K to the solidus temperature of 1700 K [14-59].

continues on next page

Table 14.8.1 – continued from previous page

SAS4A Input Location	Fortran Variable	Symbol	Describing Section	Equation Number	Suggested Value in MKS Units	Comments
1198	RHSLBT	$\rho_{Nl,ls}$	14.4.6.3	Eq. (14.4-226), Eq. (14.4-228)	728.0	This is liquid sodium density at about 1200 K [14-61].
1199	RHSLTP	$\rho_{Nl,us}$	14.4.6.3	Eq. (14.4-227), Eq. (14.4-228)	0.975 * RHSLBT	It is assumed that the temperature difference between the upper and lower liquid coolant slugs is about half of the coolant temperature rise in core (ΔT_{core}), and the corresponding density difference is 2.5% for (ΔT_{core}) of about 150K. The effect of the transient heat-up of the coolant on this estimate of the input has been ignored.
1201	CO-EFDL(2)				9.3D-5	Ref. [14-59].
1210	EGSESO		14.5.2	Eq. (14.5-10)	0.834D6	Ref. [14-59].
1211	EGSELQ		14.5.2	Eq. (14.5-10)	EGSESO + UEMELT(1	This value is obtained by adding the latent heat of fusion of cladding type 1 (location 816) to the solidus internal energy.
1212	CPSE		14.5.2		774.0	This is the constant value of the cladding specific heat at temperatures above the liquidus [14-59].

continues on next page

Table 14.8.1 – continued from previous page

SAS4A Input Location	Fortran Variable	Symbol	Describing Section	Equation Number	Suggested Value in MKS Units	Comments
1214	FNSROS				0.2	The structure thickness should be divided into two mesh intervals such that the mesh interval contacting the coolant has a thickness equal to 20% of the total thickness. For consistency, this parameter must be in agreement with the input structure mesh thicknesses in **DSTIZ** and **DSTOZ**.
1231	SRFMLE				1.0	Also see LEVITATE input description in Chapter 16.
Block 51, INPCHN						
71	NRPI1				NPIN	All pins are assumed to fail when the fuel pin failure criterion is satisfied.
74	IPSIZE				1	
75	IBUGPL				0	
76	ICFINE				0	
77	IPRINT				0	
78	IPLOT				2	
79	IBGO				0	
80	IBSTOP				0	
81	IBNEW				0	
82	IPGO				0	
83	IPSTOP				2000	
84	IPNEW				20	
Block 65, FUELIN						
2	FMELTM		14.1.2, *14.7.1*		0.2	This is required for the in-pin flow calculation of molten fuel.

continues on next page

Table 14.8.1 – continued from previous page

SAS4A Input Location	Fortran Variable	Symbol	Describing Section	Equation Number	Suggested Value in MKS Units	Comments
19	PFFAIL				0.0	This value of PCFAIL implies that the DEFORM-IV computed molten fuel cavity pressure at pin failure time will be used in the PLUTO2 calculation.

The PLUTO2 variables that are printed in output items 1 and 2 of Figure 14.8.1 are listed below:

ISTEP	Current main (point kinetics) time-step number.
TOTAL TIME	Time since beginning of transient. This is the regular SAS4A time.
ICYCLE	Current PLUTO2 calculational cycle for the SAS4A channel shown at the top for the page; the calculational step for the cycle number shown has just been completed at the time of a printout (see flowchart in Figure 14.7.3).
TIMEPL	PLUTO2 time. This is counted from the time of pin failure in this channel.
DTPLU	Current PLUTO2 time-step size.
In output item 3 total masses per subassembly are shown:	
SMFUPI	Current total mass of fuel in all failed pins of one subassembly.
SMFUST	Current total stationary (unmelted) fuel mass in all failed pins of one subassembly.
SMFUCA	Current total molten fuel mass in the pin cavities of all failed pins of one subassembly.
SMFICA	Current total free fission-gas mass in the pin cavities of all failed pins of one subassembly.
SMFSCA	Current total dissolved fission-gas mass in the cavities of all failed pins.
In output item 4, various other total masses for one subassembly are displayed:	
SMFUME	Total mass of fuel which has melted into the molten pin cavities of all failed pins in one subassembly since the time of pin failure.

continues on next page

Table 14.8.2 – continued from previous page

ISTEP	Current main (point kinetics) time-step number.
SMFIME	Total mass of the free fission gas having been added to the cavities of all failed pins in one subassembly since the time of pin failure; the total mass of dissolved fission gas having been added t the pin cavities can be calculated by multiplying SMFIME by (1-FNFIGB)/FNFIGB where FNFIGB is the input fraction of fission gas on the grain boundaries of solid fuel.
SMFSRT	Total mass of free fission gas that has been generated due to coalescence of the small (dissolved) fission-gas bubbles in all the cavities of the failed pins of one subassembly.
SMFUEJ	Total mass of fuel that has been ejected into the coolant channels since pin failure. The sum of SMFUEJ and SMFUPI should always give the total initial fuel mass of the failed pins in the subassembly considered.
SMFIEJ	Total free fission-gas mass ejected from all failed pins per subassembly.
In output item 5, several columns showing various quantities per single pin node of failure group 1 are printed:	
K	Axial cell index of the pin grid; only cells which cover the molten cavity are shown. The pin gird actually extends from the bottom of the lower blanket to the top of the upper blanket.
IDISR	Gives information about cladding disruption and fuel and gas ejection. IF IDISR = 0, cladding is not ruptured; if IDISR = 2, cladding has ruptured but currently there is no ejection; if IDIST = 3, cladding has ruptured and fuel and gas ejections are going on.
DICA	Current diameter of the molten cavity in pin cell K.
AREAFR	Current are fraction of the molten pin cavity in pin cell K (i.e., cavity cross section/total fuel cross section).
FUSTAT	Stationary (solid) fuel mass in cell K. Stationary core fuel masses above and below the cavity are shown in output item 16.
FUELM	Mobile (cavity) fuel mass in cell K.
FUELSD	Fuel smear density in the cavity.
RHFUCA	Theoretical fuel density in the cavity, the fuel volume fraction in the cavity is the ratio of FUELSD to RHFUCA.
FISGM	Free fission-gas mass in the cavity part of cell K.
FISGDM	Dissolved fission-gas mass in the cavity part of cell K.

continues on next page

Table 14.8.2 – continued from previous page

ISTEP	Current main (point kinetics) time-step number.
FNFIGB	Fraction of fission gas on the grain boundaries of the melting-in fuel; this fraction is instantaneously becoming free fission gas upon melt-in; it is presently a single input value.
FUEJ	Fuel mass ejected from pin cell K during the current time step.
FIEJ	Free fission-gas mass ejected from pin cell K during the current time step.
Output item 6 also prints several columns displaying variables pertaining to the in-pin motion:	
K	Axial cell index of the pin grid.
I	Axial index of the corresponding channel cell.
ZZPI	Axial location of the lower cell boundary in the pin cavity.
UFPI	Fuel/fission-gas mixture velocity at the lower cell boundary. All other pin cavity quantities shown are at the cell centers.
PRCA	Total physical pressure in-cavity cell K.
PRFVPI	Fuel vapor pressure component that is based on the average fuel temperature in cavity cell K.
PRVI	Artificial viscous pressure for damping numerical oscillations behind shock fronts.
EGFUCA	Average internal fuel energy in cavity cell K.
TEFUCA	Average fuel temperature in cavity cell K.
FUMESM	Mass of fuel that has molten into cavity cell K during the current time step.
FIMESM	Mass of free fission gas that has been added to cell K during the current time step due to fuel melt-in.
SIGCL	Cladding hoop stress for cell K; a negative value indicates that the channel pressure is higher than the cavity pressure. A value of zero indicates that the cladding in this cell has failed.
UTS	Ultimate tensile strength of the cladding material at temperature TECLIN (see output item 15).
Output item 7 shows the sum of the fuel and sodium voiding reactivities and also these two reactivities separately. These reactivities are for the entire SAS4A channel under consideration which usually contains more than one subassembly (see input quantity NSUBAS). For multi-channel runs the same fuel and sodium voiding reactivities are also printed out in the short reactivity printout which appears at the end of each main (point kinetics) time step.	
Output item 8 is the first row describing coolant channel variables; it prints several important integer quantities in the coolant channels (see also Figure 14.1.4):	

continues on next page

Table 14.8.2 – continued from previous page

ISTEP	Current main (point kinetics) time-step number.
IFMIBT	Bottom cell of the interaction region (i.e., the region between the upper and lower sodium slugs).
IFMITP	Top cell of the interaction region.
IFFUBT	Bottom cell of the fuel region.
IFFUTP	Top cell of the fuel region.
IFFIBT	Bottom cell of fission-gas region.
IFFITP	Top cell of fission-gas region.
IFFVBT	Bottom cell of fuel vapor region.
IFFVTP	Top cell of fuel vapor region.
IFRIBT	Lowest cell containing a cladding rupture.
IFRITP	Highest cell containing a cladding rupture.
PRIN	Current inlet pressure.
PREX	Current outlet pressure.
Output item 9 shows total fuel and sodium masses in the interaction region and the fuel and sodium masses which have been ejected out of the top of the subassembly:	
TOFUMA	Total fuel mass in the coolant channels of one subassembly; this should be equal to SMFUEJ (see output item4) until fuel gets ejected out of the top of the subassembly.
FUMATP	Fuel mass ejected out of the top of one subassembly.
TONAMA	Total sodium mass in the interaction region of one subassembly.
TPNAMA	Sodium mass ejected out of the top of one subassembly.
Output item 10 shows total fission-gas masses:	
TOFIMA	Total mass of free fission-gas in the interaction region of one subassembly.
TODGCH	Total dissolved fission-gas mass in the interaction region of one subassembly.
TODGCL	Total dissolved fission-gas mass in the plated-out fuel on the cladding and structure (this is included in TODGCH discussed above).
FIMATP	Total free and dissolved gas masses ejected out of the top of one subassembly.
Output item 11 also shows total fission-gas masses per subassembly:	
TOFIDG	Total free and dissolved fission-gas masses in all subchannels and in all failed pins of one subassembly.
TODG	Total dissolved fission-gas in all subchannels and in the cavities of all failed pins of one subassembly.
TOFI	Total free fission gas in the cavities of all failed pins and in all subchannels of one subassembly.

continues on next page

Table 14.8.2 – continued from previous page

ISTEP	Current main (point kinetics) time-step number.
TOFIST	Total fission-gas mass in the stationary (unmelted) fuel of all failed pins in a subassembly (only one type of gas is currently treated in the solid part of the fuel pins in SAS4A).
Output item 12 shows channel interface locations and velocities of the various component regions (see Figure 14.1.4 in Section 14.1):	
SLIFBT(1)	Location of the upper slug interface of the lower coolant slug, which is also the location of the lower interaction zone boundary.
SLIFBR(2)	Velocity of the upper slug interface of the lower coolant slug.
SLIFTP(1)	Location of the lower slug interface of the upper coolant slug.
SLIFTP(2)	Velocity of the lower slug interface of the upper coolant slug.
FUIFBT(1)	Location of the lower boundary of the region containing fuel.
FUIFBT(2)	Velocity of the lower boundary of the region containing fuel.
FUIFTP(1)	Location of the upper boundary of the region containing fuel.
FUIFTP(2)	Velocity of the upper boundary of the region containing fuel.
FIIFBT(1)	Location of the lower boundary of the region containing free fission gas.
FIIFTP(1)	Location of the upper boundary of the region containing free fission gas.
Output item 13 shows several columns of variables pertaining to individual channels cells:	
I	Axial channel index.
K	Index of pin cavity cell which is at the same elevation as channel cell I.
ZC	Location of the lower boundary of cell I.
UMCH	Velocity of the mixture of liquid sodium, sodium vapor, free fission gas and fuel vapor at the lower boundary of cell I (i.e., at ZC).
UFCH	Velocity of the liquid or solid fuel at the lower boundary of cell I (i.e., at ZC).
PRCH	Total pressure in cell I.

continues on next page

Table 14.8.2 – continued from previous page

ISTEP	Current main (point kinetics) time-step number.
PRNV	Sodium vapor pressure which can be: 10^{-2} which indicates that the sodium vapor pressure is being suppressed due to a sodium liquid-phase pressure. Sodium saturation pressure if liquid is still present in cell I (i.e., THNL greater than zero - see output item 14). Gas pressure of superheated sodium vapor if no liquid sodium is left in channel cell I (i.e. if THNL is equal to zero - see output time 13).
PRFI	Free fission-gas pressure. This pressure component is equal to the total pressure for a sodium single-liquid-phase pressure situation.
PFFV	Fuel vapor pressure.
THCHOP	Open channel cross-sectional area per subassembly divided by reference area AXMS which is input. The open channel cross-sectional area is the total channel cross-section area minus the cross-sectional area of the frozen fuel crusts. THCHOP can also be interpreted as the open channel volume fraction.
THNAFM	Liquid sodium film cross-sectional area per subassembly divided by the input reference area AXMX.
THNL	Total liquid sodium cross-sectional area per subassembly divided by the input reference area AXMX.
Output item 14 contains the other area fractions and also the total component masses per axial cell I:	
I	Axial channel index.
IFLAG	Indicates the type of fuel flow regime present in cell I. IFLAG = 1 indicates the particulate flow regime, IFLAG = 3 indicates partial or full annular fuel flow, and IFLAG = 4 indicates bubbly fuel flow;
THFUCH	Mobile channel fuel cross-sectional area per subassembly divided by the input reference area AXMX.
THFF	Stationary fuel crust cross-sectional area per subassembly divided by the input reference area AXMX.
THVG	Cross-sectional area of the vapor/gas flow per subassembly divided by input reference area AXMX.
FUMASS	Mobile channel fuel mass in channel cell I (a channel cell in PLUTO2 includes all subchannels of a subassembly).
FUFFMA	Stationary frozen film mass in channel cell I.
NAMASS	Total sodium mass in channel cell I.
FIMASS	Total free fission-gas mass in channel cell I.

continues on next page

Table 14.8.2 – continued from previous page

ISTEP	Current main (point kinetics) time-step number.
FIDGMA	Mass of fission gas dissolved in the mobile fuel in channel cell I.
FIFFMA	Mass of fission gas dissolved in the frozen fuel crusts in channel cell I.
Output item 15 contains the variable columns giving channel, cladding, and structure temperatures in cell I:	
I	Axial channel index.
IDISR	Gives information about cladding disruption and fuel and gas ejection: If IDISR = 0, cladding is not ruptured; if IDISR = 2, cladding is ruptured but currently no ejection going on; if IDISR = 3, cladding is ruptured and fuel and gas ejection are going on; IDISR also appears in output item 5.
TENA	Sodium temperature.
TEFUOS	Fuel temperature. The fuel temperature will be in the melting range if TFSOL < TEFUOS < TFLIQ (TFSIK and TFLIQ ARE INPUT). The fuel melt fraction in this case is: (TEFUOS - TFSOL)/(TFLIQ - TFSOL);
TEFFCL	Temperature of the frozen fuel crust on the cladding.
TEFFSR	Temperature of the frozen fuel crust on the structure.
TECLOS	Temperature of the cladding outer surface. The outer cladding temperature will be in the melting range if TESOL < TECLOS < TELIQ (TESOL and TELIQ are input). The cladding melt fraction in this case is: (TECLOS - TESOL)/(TELIQ - TESOL);
TECLIN	Temperature of the middle cladding node.
TESROS	Temperature of the structure surface facing the coolant channel.
TESRIN	Temperature of the structure node facing the neighboring subassembly.
Output items 16 and 17 show the pin and total fuel masses for the entire axial region containing fuel. The fuel-pin masses outside the interaction region were not shown before. The following variables are displayed:	
MAFUPI	Total mass in all fuel-pin cells corresponding to channel cell I (including the stationary fuel in both the failed and unfailed pin nodes and the mobile fuel in the failed pin nodes).
MAFUTO	Sum of the total fuel in channel cell I and MAFUPI (see above). This output is valuable for comparing with one-dimensional hodoscope fuel distribution curves.

continues on next page

Table 14.8.2 – continued from previous page

ISTEP	Current main (point kinetics) time-step number.
Output item 18 contains the variable columns pertaining to the temperature distribution along the entire channel including the regions outside the interaction region: It should be noted that the temperatures outside the interaction region which are shown here are calculated at the end of the last heat-transfer time step which can be fractions of a millisecond before the time of the current printout.	
I	Axial channel index.
ZCOOL	Location of the lower boundary of mesh cell I.
TREFL2(2)	Inner reflector node temperature. Reflectors can be located only below or above the pin zone K2PIN.
TREFL2(1)	Temperature of the outer reflector node which is facing the coolant.
T1(NEPP)	Inner cladding surface temperature (i.e., next to the fuel).
T1(NE)	Cladding temperature of the middle cladding node.
T1(NEP)	Outer cladding surface temperature (i.e., next to the coolant).
TENA	Sodium temperature.
TSAT	Sodium saturation temperature. This is calculated only for the pin zone to detect sodium-boiling initiation.
TSTR2(1)	Temperature of the structure node facing the coolant channel.
TSTR2(2)	Temperature of the structure node facing the neighboring hexcan wall.
PRCH	Pressure in the coolant channel. This is calculated outside the fuel-pin zone only if the interaction region extends beyond it.
Output item 18 shows the message “PLUTO ENDS”. This indicates that control is transferred from PLUTO2 to the SAS4A transient driver TSTHRM at the end of a primary-loop time step. The variables printed after the message “PLUTO ENDS” are not labeled in order to save space in multi-channel runs. These variables are:	
ICYCLE	PLUTO2 calculational cycle.
IFMIBT	Lowermost interaction zone cell.
IFMITP	Uppermost interaction zone cell.
IFRIBT	Lowermost channel cell with failed cladding.
IFRITP	Uppermost cladding cell with failed cladding.
IFFUBT	Lowermost channel cell containing fuel.
IFFUTP	Uppermost channel cell containing fuel.
ICH	SAS4A calculational channel.
REA1	Sodium voiding reactivity for channel ICH.
REA2	Fuel reactivity for channel ICH.

continues on next page

Table 14.8.2 – continued from previous page

ISTEP	Current main (point kinetics) time-step number.
TIMEPL	PLUTO2 time for channel which is counted from the time of pin failure.

The above 11 variables of output item 18 are also printed with the message “PLUTO STARTS”, in the same order, at the time of entry to the PLUTO2 module, before printing output item 1.

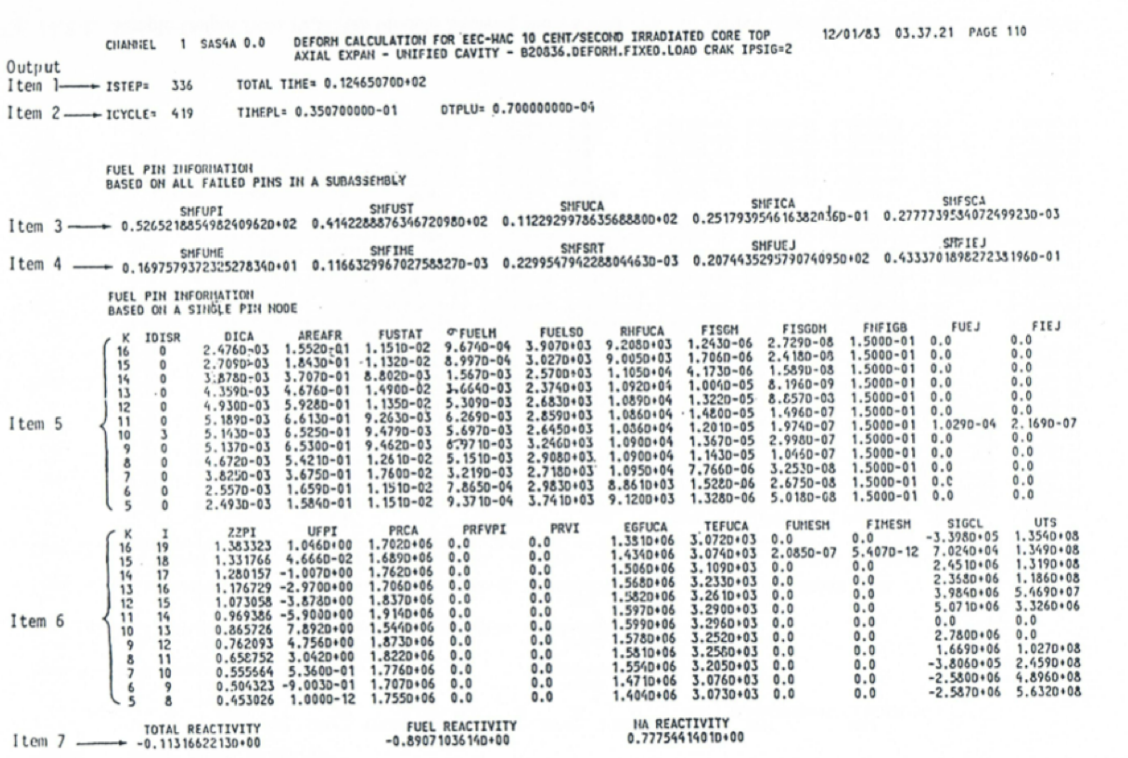


Figure 14.8.1: Sample Output from PLUTO2 Module showing the Various Items of Output Described in Section 14.8.2

14.9 References

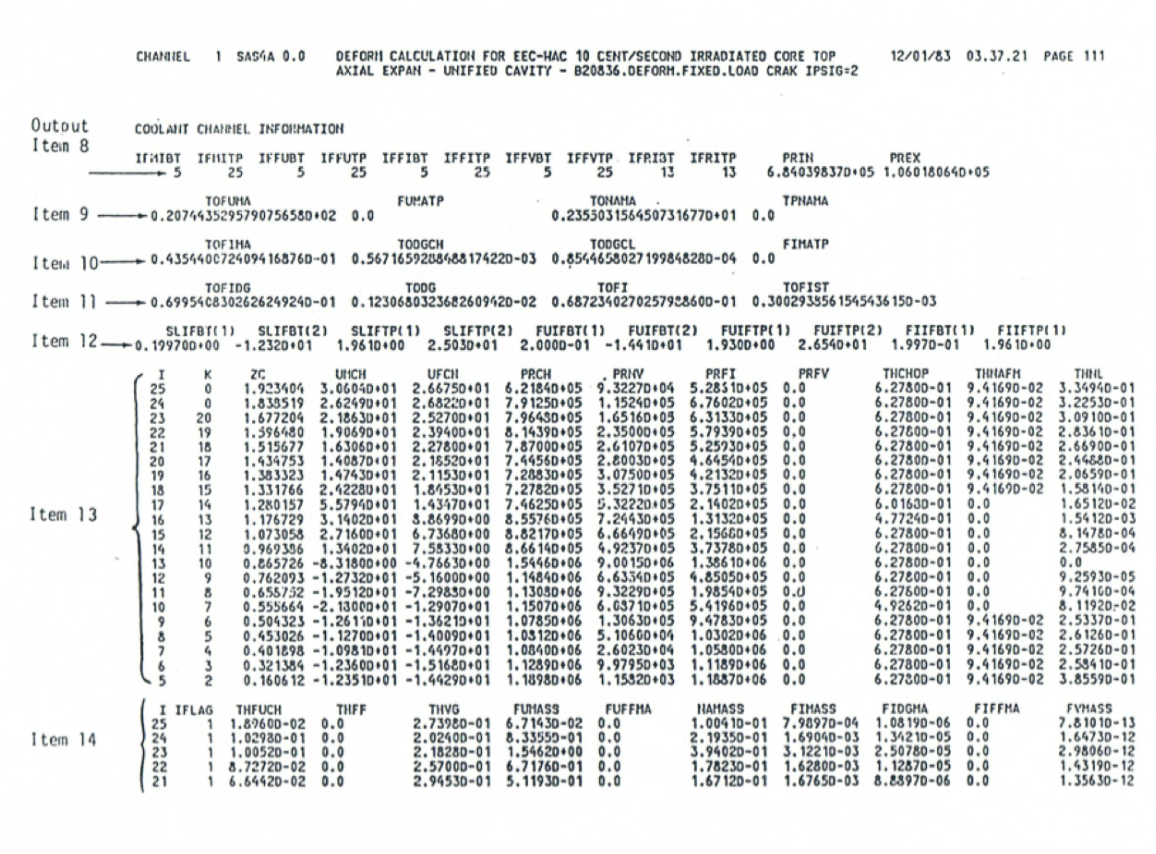
14-1. L. L. Smith, et al., Unpublished information, Argonne National Laboratory, 1975.

14-2. W. R. Bohl, Unpublished information, Argonne National Laboratory, 1974.

14-3. H. U. Wider, “An Improved Analysis of Fuel Motion During an Overpower Excursion,” Ph.D. Thesis, Northwestern University, Evanston, Illinois, June 1974.

14-4. H. U. Wider, et al., “An Improved Analysis of Fuel Motion During an Overpower Excursion,” *Fast Reactor Safety*, CONF-740401-P3, Beverly Hills, California, p. 1541, April 1974.

14-5. A. B. Rothman, et al., “Results of Recent TOP and LOF Experiments in TREAT,” *Proc. Int. Melting on Fast Reactor Safety and Related Physics*, Chicago, Illinois, CONF-761001, p. 1625, October 1976.



```
CHANNEL   1  SAS4A 0.0    DEFORM CALCULATION FOR EEC-HAC 10 CENT/SECOND IRRADIATED CORE TOP      12/01/83  03.37.21  PAGE 111
                          AXIAL EXPAN - UNIFIED CAVITY - B20836.DEFORM.FIXED.LOAD CRAK IPSIG=2

Outout   COOLANT CHANNEL INFORMATION
Item 8
      IFMIBT  IFMITP  IFFUBT  IFFUTP  IFFIBT  IFFITP  IFFVBT  IFFVTP  IFRIBT  IFRITP      PRIN          PREX
          5      25       5      25       5      25       5      25      13      13   6.84039837D+05 1.06018064D+05

Item 9          TOFUMA                  FUMATP                  TONAMA                  TPNAMA
     ---->  0.20744352957907565D+02  0.0                     0.23550315645073167D+01  0.0

Item 10         TOFIMA                  TODGCH                  TODGCL                  FIMATP
     ---->  0.43544007240941687 6D-01  0.56716592884881742 2D-03  0.85446580271998482 80-04  0.0

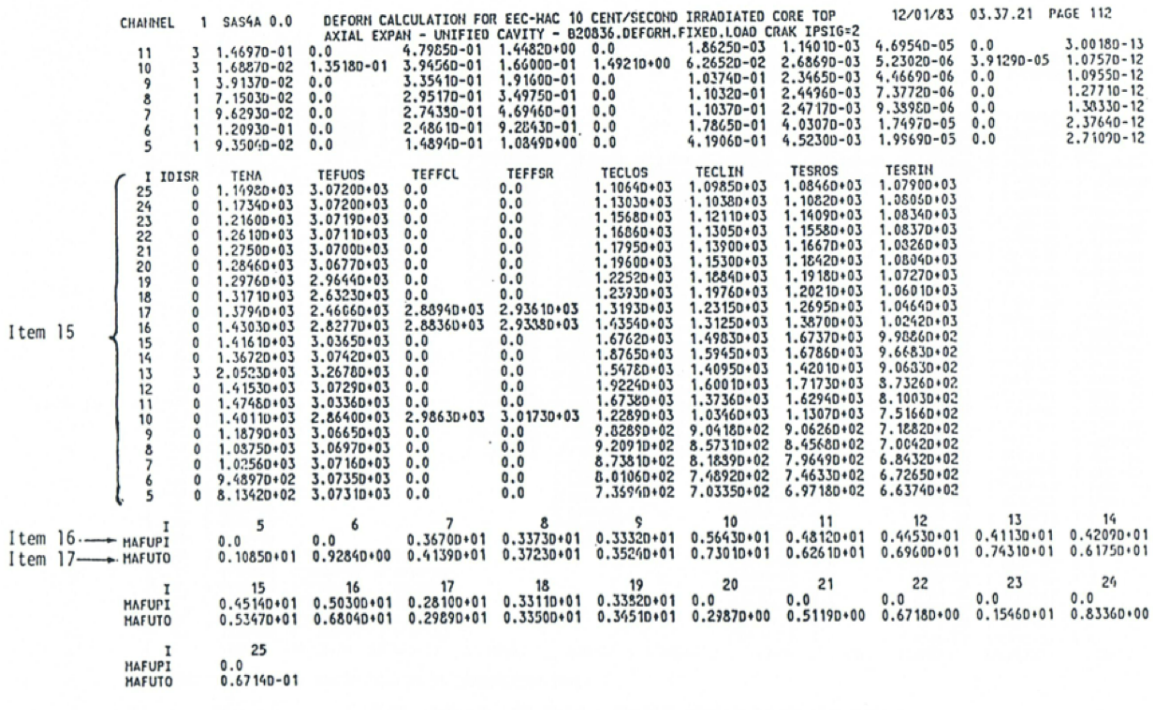
Item 11         TOFIDG                  TODG                    TOFI                    TOFIST
     ---->  0.69954083026262492 4D-01  0.12306803236826094 2D-02  0.68723402702579286 0D-01  0.30029385615454361 5D-03

Item 12         SLIFBT(1)   SLIFBT(2)   SLIFTP(1)   SLIFTP(2)   FUIFBT(1)   FUIFBT(2)   FUIFTP(1)   FUIFTP(2)   FIIFBT(1)   FIIFTP(1)
     ---->  0.19970D+00  -1.232D+01   1.961D+00   2.503D+01   2.000D-01  -1.441D+01   1.930D+00   2.654D+01   1.997D-01   1.961D+00

Item 13
  I   K    ZC        UHCH        UFCH        PRCH        PRNV        PRFI        PRFV       THCHOP      THNAFM      THNL
 25   0   1.923404  3.0604D+01  2.6675D+01  6.2184D+05  9.3227D+04  5.2851D+05  0.0        6.2780D-01  9.4169D-02  3.3494D-01
 24   0   1.838519  2.6249D+01  2.6822D+01  7.9125D+05  1.1524D+05  6.7602D+05  0.0        6.2780D-01  9.4169D-02  3.2253D-01
 23  20   1.677204  2.1863D+01  2.5270D+01  7.9643D+05  1.6516D+05  6.3133D+05  0.0        6.2780D-01  9.4169D-02  3.0910D-01
 22  19   1.596480  1.9069D+01  2.3940D+01  8.1439D+05  2.3500D+05  5.7939D+05  0.0        6.2780D-01  9.4169D-02  2.8361D-01
 21  18   1.515677  1.6306D+01  2.2780D+01  7.8700D+05  2.6107D+05  5.2593D+05  0.0        6.2780D-01  9.4169D-02  2.6690D-01
 20  17   1.434753  1.4087D+01  2.1852D+01  7.4456D+05  2.8003D+05  4.6454D+05  0.0        6.2780D-01  9.4169D-02  2.4488D-01
 19  16   1.383323  1.4743D+01  2.1153D+01  7.2883D+05  3.0750D+05  4.2132D+05  0.0        6.2780D-01  9.4169D-02  2.0659D-01
 18  15   1.331766  2.4228D+01  1.8653D+01  7.2782D+05  3.5271D+05  3.7511D+05  0.0        6.2780D-01  9.4169D-02  1.5814D-01
 17  14   1.280157  5.5794D+01  1.4347D+01  7.4625D+05  5.3222D+05  2.1402D+05  0.0        6.0163D-01  0.0         1.6512D-02
 16  13   1.176729  3.1402D+01  8.8699D+00  8.5576D+05  7.2443D+05  1.3132D+05  0.0        4.7724D-01  0.0         1.5412D-03
 15  12   1.073058  2.7160D+01  6.7368D+00  8.8217D+05  6.6649D+05  2.1566D+05  0.0        6.2780D-01  0.0         8.1478D-04
 14  11   0.969356  1.3402D+01  7.5833D+00  8.6614D+05  4.9237D+05  3.7378D+05  0.0        6.2780D-01  0.0         2.7585D-04
 13  10   0.865726 -8.3180D+00 -4.7663D+00  1.5446D+06  9.0015D+06  1.3861D+06  0.0        6.2780D-01  0.0         0.0
 12   9   0.762093 -1.2732D+01 -5.1600D+00  1.1484D+06  6.6334D+05  4.8505D+05  0.0        6.2780D-01  0.0         9.2593D-05
 11   8   0.658732 -1.9512D+01 -7.2983D+00  1.1308D+06  9.3229D+05  1.9854D+05  0.0        6.2760D-01  0.0         9.7410D-04
 10   7   0.555664 -2.1300D+01 -1.2907D+01  1.1507D+06  6.0371D+05  5.4196D+05  0.0        4.9262D-01  0.0         8.1192D-02
  9   6   0.504323 -1.2616D+01 -1.3621D+01  1.0785D+06  1.3063D+05  9.4783D+05  0.0        6.2780D-01  9.4169D-02  2.5337D-01
  8   5   0.453026 -1.1270D+01 -1.4009D+01  1.0512D+06  5.1060D+04  1.0302D+06  0.0        6.2780D-01  9.4169D-02  2.6126D-01
  7   4   0.401898 -1.0981D+01 -1.4497D+01  1.0840D+06  2.6023D+04  1.0580D+06  0.0        6.2780D-01  9.4169D-02  2.5726D-01
  6   3   0.321384 -1.2360D+01 -1.5168D+01  1.1289D+06  9.9795D+03  1.1189D+06  0.0        6.2780D-01  9.4169D-02  2.5841D-01
  5   2   0.160612 -1.2351D+01 -1.4429D+01  1.1898D+06  1.1582D+03  1.1887D+06  0.0        6.2780D-01  9.4169D-02  3.8559D-01

Item 14
  I  IFLAG  THFUCH      THFF        THVG        FUMASS      FUFFMA      NAMASS      FIMASS      FIDGMA      FIFFMA      FVMASS
 25      1  1.8960D-02  0.0         2.7398D-01  6.7143D-02  0.0         1.0041D-01  7.9897D-04  1.0819D-06  0.0         7.8101D-13
 24      1  1.0298D-01  0.0         2.0240D-01  8.3355D-01  0.0         2.1935D-01  1.6904D-03  1.3421D-05  0.0         1.6473D-12
 23      1  1.0052D-01  0.0         2.1828D-01  1.5462D+00  0.0         3.9402D-01  3.1221D-03  2.5078D-05  0.0         2.9806D-12
 22      1  8.7272D-02  0.0         2.5700D-01  6.7176D-01  0.0         1.7823D-01  1.6280D-03  1.1287D-05  0.0         1.4319D-12
 21      1  6.6442D-02  0.0         2.9453D-01  5.1193D-01  0.0         1.6712D-01  1.6765D-03  8.8897D-06  0.0         1.3563D-12
```

Figure 14.8.2: Sample Output from PLUTO2 Module showing the Various Items of Output Described in Section 14.8.2 (Cont'd)



```
 CHANNEL   1  SAS4A 0.0    DEFORM CALCULATION FOR EEC-HAC 10 CENT/SECOND IRRADIATED CORE TOP      12/01/83  03.37.21  PAGE 112
                           AXIAL EXPAN - UNIFIED CAVITY - B20836.DEFORM.FIXED.LOAD CRAK IPSIG=2
    11     3  1.4697D-01  0.0          4.7985D-01  1.4482D+00  0.0          1.8625D-03  1.1401D-03  4.6954D-05  0.0          3.0018D-13
    10     3  1.6887D-02  1.3518D-01   3.9456D-01  1.6600D-01  1.4921D+00   6.2652D-02  2.6869D-03  5.2302D-06  3.9129D-05   1.0757D-12
     9     1  3.9137D-02  0.0          3.3541D-01  1.9160D-01  0.0          1.0374D-01  2.3465D-03  4.4669D-06  0.0          1.0955D-12
     8     1  7.1503D-02  0.0          2.9517D-01  3.4975D-01  0.0          1.1032D-01  2.4496D-03  7.3772D-06  0.0          1.2771D-12
     7     1  9.6293D-02  0.0          2.7433D-01  4.6946D-01  0.0          1.1037D-01  2.4717D-03  9.3398D-06  0.0          1.3833D-12
     6     1  1.2093D-01  0.0          2.4861D-01  9.2843D-01  0.0          1.7865D-01  4.0307D-03  1.7497D-05  0.0          2.3764D-12
     5     1  9.3504D-02  0.0          1.4894D-01  1.0849D+00  0.0          4.1906D-01  4.5230D-03  1.9969D-05  0.0          2.7109D-12

Item 15
      I IDISR    TENA        TEFUOS      TEFFCL      TEFFSR      TECLOS      TECLIN      TESROS      TESRIN
     25     0  1.1498D+03  3.0720D+03  0.0         0.0         1.1064D+03  1.0985D+03  1.0846D+03  1.0790D+03
     24     0  1.1734D+03  3.0720D+03  0.0         0.0         1.1303D+03  1.1038D+03  1.0882D+03  1.0805D+03
     23     0  1.2160D+03  3.0719D+03  0.0         0.0         1.1568D+03  1.1211D+03  1.1409D+03  1.0834D+03
     22     0  1.2610D+03  3.0711D+03  0.0         0.0         1.1686D+03  1.1305D+03  1.1558D+03  1.0837D+03
     21     0  1.2750D+03  3.0700D+03  0.0         0.0         1.1795D+03  1.1390D+03  1.1667D+03  1.0826D+03
     20     0  1.2846D+03  3.0677D+03  0.0         0.0         1.1960D+03  1.1530D+03  1.1842D+03  1.0804D+03
     19     0  1.2976D+03  2.9644D+03  0.0         0.0         1.2252D+03  1.1884D+03  1.1918D+03  1.0727D+03
     18     0  1.3171D+03  2.6323D+03  0.0         0.0         1.2393D+03  1.1976D+03  1.2021D+03  1.0601D+03
     17     0  1.3794D+03  2.4666D+03  2.8894D+03  2.9361D+03  1.3193D+03  1.2315D+03  1.2695D+03  1.0464D+03
     16     0  1.4303D+03  2.8277D+03  2.8836D+03  2.9338D+03  1.4354D+03  1.3125D+03  1.3870D+03  1.0242D+03
     15     0  1.4161D+03  3.0365D+03  0.0         0.0         1.6762D+03  1.4983D+03  1.6737D+03  9.9886D+02
     14     0  1.3672D+03  3.0742D+03  0.0         0.0         1.8765D+03  1.5945D+03  1.6786D+03  9.6683D+02
     13     3  2.0523D+03  3.2678D+03  0.0         0.0         1.5478D+03  1.4095D+03  1.4201D+03  9.0633D+02
     12     0  1.4153D+03  3.0729D+03  0.0         0.0         1.9224D+03  1.6001D+03  1.7173D+03  8.7326D+02
     11     0  1.4748D+03  3.0336D+03  0.0         0.0         1.6738D+03  1.3736D+03  1.6294D+03  8.1003D+02
     10     0  1.4011D+03  2.8640D+03  2.9863D+03  3.0173D+03  1.2289D+03  1.0346D+03  1.1307D+03  7.5166D+02
      9     0  1.1879D+03  3.0665D+03  0.0         0.0         9.8289D+02  9.0418D+02  9.0626D+02  7.1882D+02
      8     0  1.0375D+03  3.0697D+03  0.0         0.0         9.2091D+02  8.5731D+02  8.4568D+02  7.0042D+02
      7     0  1.0256D+03  3.0716D+03  0.0         0.0         8.7381D+02  8.1839D+02  7.9649D+02  6.8432D+02
      6     0  9.4897D+02  3.0735D+03  0.0         0.0         8.0106D+02  7.4892D+02  7.4633D+02  6.7265D+02
      5     0  8.1342D+02  3.0731D+03  0.0         0.0         7.3594D+02  7.0335D+02  6.9718D+02  6.6374D+02

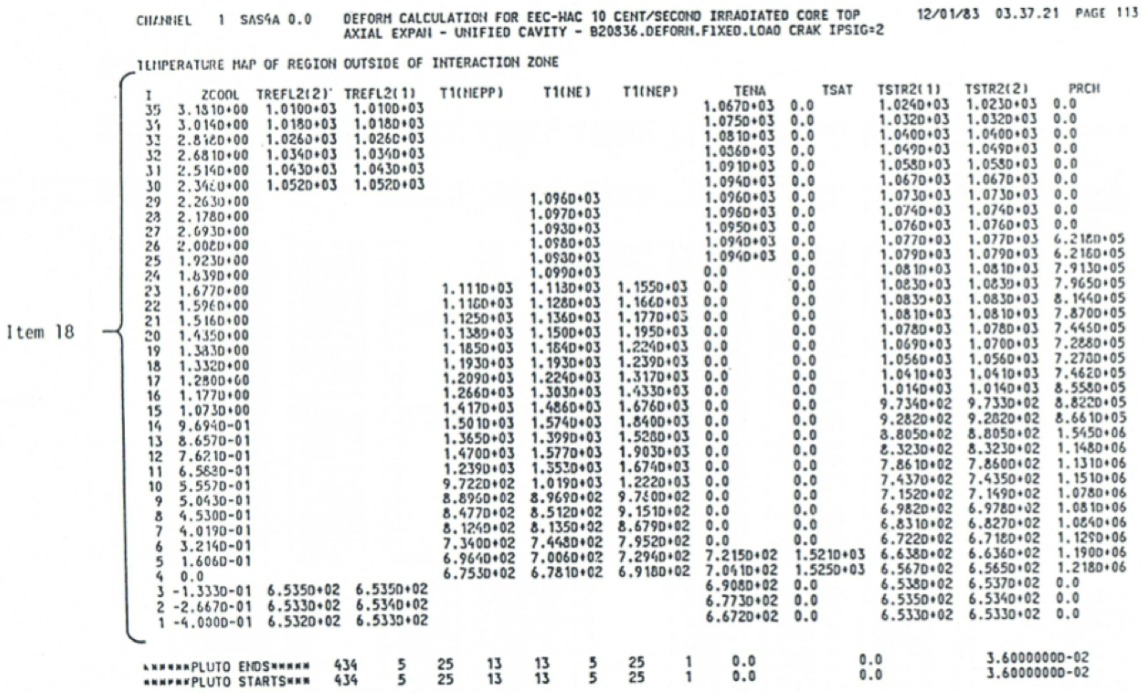
Item 16 ──► MAFUPI
Item 17 ──► MAFUTO

      I          5           6           7           8           9          10          11          12          13          14
MAFUPI      0.0         0.0         0.3670D+01  0.3373D+01  0.3332D+01  0.5643D+01  0.4812D+01  0.4453D+01  0.4113D+01  0.4209D+01
MAFUTO      0.1085D+01  0.9284D+00  0.4139D+01  0.3723D+01  0.3524D+01  0.7301D+01  0.6261D+01  0.6960D+01  0.7431D+01  0.6175D+01

      I         15          16          17          18          19          20          21          22          23          24
MAFUPI      0.4514D+01  0.5030D+01  0.2810D+01  0.3311D+01  0.3382D+01  0.0         0.0         0.0         0.0         0.0
MAFUTO      0.5347D+01  0.6804D+01  0.2989D+01  0.3350D+01  0.3451D+01  0.2987D+00  0.5119D+00  0.6718D+00  0.1546D+01  0.8336D+00

      I          25
MAFUPI      0.0
MAFUTO      0.6714D-01
```

Figure 14.8.3: Sample Output from PLUTO2 Module showing the Various Items of Output Described in Section 14.8.2 (Cont'd)



CHANNEL 1 SAS4A 0.0 DEFORM CALCULATION FOR EEC-HAC 10 CENT/SECOND IRRADIATED CORE TOP 12/01/83 03.37.21 PAGE 113
AXIAL EXPAN - UNIFIED CAVITY - B20836.DEFORM.FIXED.LOAD CRAK IPSIG=2

TEMPERATURE MAP OF REGION OUTSIDE OF INTERACTION ZONE

Item 18

I	ZCOOL	TREFL2(2)	TREFL2(1)	T1(NEPP)	T1(NE)	T1(NEP)	TENA	TSAT	TSTR2(1)	TSTR2(2)	PRCH
35	3.1810+00	1.0100+03	1.0100+03				1.0670+03	0.0	1.0240+03	1.0230+03	0.0
34	3.0140+00	1.0180+03	1.0180+03				1.0750+03	0.0	1.0320+03	1.0320+03	0.0
33	2.8460+00	1.0260+03	1.0260+03				1.0810+03	0.0	1.0400+03	1.0400+03	0.0
32	2.6810+00	1.0340+03	1.0340+03				1.0360+03	0.0	1.0490+03	1.0490+03	0.0
31	2.5140+00	1.0430+03	1.0430+03				1.0910+03	0.0	1.0580+03	1.0580+03	0.0
30	2.3460+00	1.0520+03	1.0520+03				1.0940+03	0.0	1.0670+03	1.0670+03	0.0
29	2.2630+00				1.0960+03		1.0960+03	0.0	1.0730+03	1.0730+03	0.0
28	2.1780+00				1.0970+03		1.0960+03	0.0	1.0740+03	1.0740+03	0.0
27	2.0930+00				1.0930+03		1.0950+03	0.0	1.0760+03	1.0760+03	0.0
26	2.0080+00				1.0980+03		1.0940+03	0.0	1.0770+03	1.0770+03	6.2180+05
25	1.9230+00				1.0930+03		1.0940+03	0.0	1.0790+03	1.0790+03	6.2160+05
24	1.8390+00				1.0990+03		0.0	0.0	1.0810+03	1.0810+03	7.9130+05
23	1.7720+00			1.1110+03	1.1130+03	1.1550+03	0.0	0.0	1.0830+03	1.0830+03	7.9650+05
22	1.5960+00			1.1160+03	1.1280+03	1.1660+03	0.0	0.0	1.0830+03	1.0830+03	8.1440+05
21	1.5160+00			1.1250+03	1.1360+03	1.1770+03	0.0	0.0	1.0810+03	1.0810+03	7.8700+05
20	1.4350+00			1.1380+03	1.1500+03	1.1950+03	0.0	0.0	1.0780+03	1.0780+03	7.4450+05
19	1.3530+00			1.1850+03	1.1840+03	1.2240+03	0.0	0.0	1.0690+03	1.0700+03	7.2880+05
18	1.3320+00			1.1930+03	1.1930+03	1.2390+03	0.0	0.0	1.0560+03	1.0560+03	7.2700+05
17	1.2880+00			1.2090+03	1.2240+03	1.3170+03	0.0	0.0	1.0410+03	1.0410+03	7.4620+05
16	1.1770+00			1.2660+03	1.3030+03	1.4330+03	0.0	0.0	1.0140+03	1.0140+03	8.5580+05
15	1.0730+00			1.4170+03	1.4860+03	1.6760+03	0.0	0.0	9.7340+02	9.7330+02	8.8220+05
14	9.6940-01			1.5010+03	1.5740+03	1.8400+03	0.0	0.0	9.2820+02	9.2820+02	8.6610+05
13	8.6570-01			1.3650+03	1.3990+03	1.5280+03	0.0	0.0	8.8050+02	8.8050+02	1.5450+06
12	7.6210-01			1.4700+03	1.5770+03	1.9030+03	0.0	0.0	8.3230+02	8.3230+02	1.1480+06
11	6.5830-01			1.2390+03	1.3530+03	1.6740+03	0.0	0.0	7.8610+02	7.8600+02	1.1310+06
10	5.5570-01			9.7220+02	1.0190+03	1.2220+03	0.0	0.0	7.4370+02	7.4350+02	1.1510+06
9	5.0430-01			8.8960+02	8.9690+02	9.7800+02	0.0	0.0	7.1520+02	7.1490+02	1.0780+06
8	4.5300-01			8.4770+02	8.5120+02	9.1510+02	0.0	0.0	6.9820+02	6.9780+02	1.0810+06
7	4.0190-01			8.1240+02	8.1350+02	8.6790+02	0.0	0.0	6.8310+02	6.8270+02	1.0840+06
6	3.2140-01			7.3400+02	7.4480+02	7.9520+02	0.0	0.0	6.7220+02	6.7180+02	1.1290+06
5	1.6060-01			6.9640+02	7.0060+02	7.2940+02	7.2150+02	1.5210+03	6.6380+02	6.6360+02	1.1900+06
4	0.0			6.7530+02	6.7810+02	6.9180+02	7.0410+02	1.5250+03	6.5670+02	6.5650+02	1.2180+06
3	-1.3330-01	6.5350+02	6.5350+02				6.9080+02	0.0	6.5380+02	6.5370+02	0.0
2	-2.6670-01	6.5330+02	6.5340+02				6.7730+02	0.0	6.5350+02	6.5340+02	0.0
1	-4.0000-01	6.5320+02	6.5330+02				6.6720+02	0.0	6.5330+02	6.5330+02	0.0

******PLUTO ENDS****** 434 5 25 13 13 5 25 1 0.0 0.0 3.6000000D-02
******PLUTO STARTS***** 434 5 25 13 13 5 25 1 0.0 0.0 3.6000000D-02

Figure 14.8.4: Sample Output from PLUTO2 Module showing the Various Items of Output Described in Section 14.8.2 (Cont'd)

14-6. H. U. Wider and L. A. Semenza, "Analysis of TREAT Transient Overpower Experiments Using the PLUTO Codes," Specialists Workshop on Predictive Analysis of Material Dynamics in LMFBR Safety Experiments, Los Alamos Scientific Laboratory Report, March 1979.

14-7. P. A. Pizzica and P. B. Abramson, "A Numerical Model of Reactor Fuel and Coolant Motions Following Pin Failure," *Nucl. Sci. Eng.*, vol. 64, p. 465-479, 1977.

14-8. P. A. Pizzica, P. L. Garner, and P. B. Abramson, "A User's Guide to EPIC, a Computer Program to Calculate the Motion of Fuel and Coolant Subsequent to Pin Failure in an LMFBR," ANL-80-47, Argonne National Laboratory, Argonne, Illinois, October 1979.

14-9. D. H. Cho, R. O. Ivins, and R. W. Wright, "Pressure Generation by Molten Fuel-Coolant Interactions Under LMFBR Accident Conditions," *Proc. Conf. on New Developments in Reactor Mathematics and Applications*, Idaho Falls, March 1971.

14-10. D. P. Weber and H. U. Wider Eds., Unpublished information, Argonne National Laboratory, 1977.

14-11. H. U. Wider, "PLUTO2: A Computer Code for the Analysis of Overpower Accidents in LMFBRs," *Trans. Am. Nucl. Soc.*, vol. 27, p. 533, November 1977.

14-12. H. U. Wider, A. M. Tentner, and P. A. Pizzica, "The PLUTO2 Overpower Excursion Code and a Comparison with EPIC," *Fast Reactor Safety Meeting*, Seattle, Washington, vol. 1, p. 120, August 1979.

14-13. R. J. Page, et al., Unpublished information, Argonne National Laboratory, 1978.

14-14. R. O. McNary, et al., Unpublished information, Argonne National Laboratory, 1979.

14-15. C. H. Bowers, H. U. Wider, and A. M. Tentner, "Analysis of TREAT Transient Tests L7 and L8 with SAS3D, LEVITATE, and PLUTO2," Specialists Workshop on Predictive Analysis of Material Dynamics in LMFBR Safety Experiments, Los Alamos Scientific Laboratory, March 1979.

14-16. A. M. Tentner, H. U. Wider, and C. H. Bowers, "A Mechanistic Model of Fuel Flow Regimes and Fuel Plateout in LMFBR Overpower Accidents," *Trans. Am. Nucl. Soc.*, vol. 30, p. 448, November 1978.

14-17. A. E. Wright, et al., Unpublished information, Argonne National Laboratory, 1978.

14-18. J, K. Bigelow, G. E. Russher, and S. E. Seeman, Unpublished information, Hanford Engineering Development Laboratory, 1980.

14-19. W. A. Ragland, et al., Unpublished information, Argonne National Laboratory, 1981.

14-20. P. C. Cacciabue, et al., "Comparative Analysis of a Hypothetical 9.1 $/sec Transient Overpower Accident in an Irradiated LMFBR Core Using Different Computer Models," EUR 8018 EN, Nuclear Science and Technology, Commission of the European Community, Luxembourg, 1982.

14-21. Argonne National Laboratory, Unpublished information, 1975.

14-22. R. B. Bird, W. E. Stewart, and E. N. Lightfoot, *Transient Phenomena*, John Wiley & Sons, Inc., New York, 1960.

14-23. R. G. Deissler, "Analysis of Turbulent Heat Transfer, Mass Transfer, and Friction in Smooth Tubes at High Prandtl an Schmidt Numbers," NACA Report 1210, 1955.

14-24. A. E. Wright, et al., "Transient Overpower Test H5 on FFTF-Type Intermediate-Power Fuel," Unpublished information, Argonne National Laboratory, Illinois, June 1975. Also see A. E. Wright, et al., "Inpile Failure Threshold Experiment (H5) With Pre-Irradiated FFTF-Type Fuel," *Trans. Am. Nucl. Soc.*, vol. 18, p. 211, June 1975.

14-25. R. J. Page A. B. Rothman, and P. H. Froehle, Unpublished information, Argonne National Laboratory, 1979.

14-26. W. L. Hu, Unpublished information, Hanford Engineering Development Laboratory, 1983.

14-27. H. U. Wider, J. f. Jackson, and D. T. Eggen, "An Improved Viscous Pressure Formulation for Two-Phase Compressible Hydrodynamics Calculations," *Trans. Am. Nucl. Soc.*, vol. 17, p. 246, November 1973.

14-28. R. E. Henry, M. A. Grolmes, and H. K. Fauske, "Pressure Pulse Propagation in Two-Phase One- and Two-Component Mixtures," ANL-7792, Argonne National Laboratory, Argonne, Illinois, March 1971.

14-29. L. L. Smith, Unpublished information, Los Alamos Scientific Laboratory, 1980.

14-30. J. H. Stuhmiller and R. E. Ferguson, "Comparison of Numerical Methods for Fluid Flows," EPRI NP-1236, Electric Power Research Institute, Palo Alto, California, November 1979.

14-31. Patrick J. Roache, *Computational Fluid Dynamics*, Hermosa Publishers, Albuquerque, New Mexico, 1976.

14-32. H. U. Wider, J. F. Jackson, and W. R. Bohl, "The Slip Flow Treatment in the PLUTO Code," *Proc. Conf. on Computational Methods in Nuclear Engineering*, Charleston, South Carolina, CONF-760413, Vol. 1, pp. II-125 to II-144, April 1975.

14-33. M. Ishii and M. A. Grolmes, "Inception Criteria for Droplet Entrainment in Two-Phase Concurrent Film Flow," *AICHE J.*, vol. 21, no. 2, p. 308-317, March 1975.

14-34. M. Ishii and M. A. Grolmes, Unpublished information, Argonne National Laboratory, 1974.

14-35. R. N. Knoopman, et al., Unpublished information, Argonne National Laboratory, 1977.

14-36. A. E. Wright, et al., Unpublished information, Argonne National Laboratory, 1976.

14-37. H. U. Wider and A. E. Wright, "Analysis of a Sodium Reentry Event in the H4 TREAT Tests," *Trans. Am. Nucl. Soc.*, vol. 22, p. 428, November 1975.

14-38. B. W. Spencer, et al., "Fuel Motion in the CAMEL TOP-Simulation LMFBR Safety Tests," *Proc. Intl. Meeting on Fast Reactor Safety*, Seattle, Washington, vol. 4, p. 1756-1765, August 1979.

14-39. Argonne National Laboratory, Unpublished information, 1980.

14-40. S. J. Board and R. W. Hall, "Recent Advances in Understanding Large Scale Vapor Explosions," *Proc. Third Specialist Meeting on Sodium/Fuel Interaction in Fast Reactors*, Tokyo, Power Reactor and Nuclear Fuel Development Corporation Report PNC N251 65-12, vol. 1, pp. 249-284, March 1976.

14-41. H. Jacobs, "Computational Analysis of Fuel-Sodium Interaction With an Improved Method," *Proc. Int. Meeting on Fast Reactor Safety and Related Physics*, Chicago, CONF-761001, pp. 926-935, October 1976.

14-42. W. A. Ragland, et al., "Interpretation of SLSF Experiment P4 Instrument Response," *Trans. Am. Nucl. Soc*., vol. 44, p. 332-333, June 1983.

14-43. Argonne National Laboratory, Unpublished information, 1978.

14-44. S. J. Hakim, Unpublished information, Los Alamos Scientific Laboratory, 1979.

14-45. M. Epstein and G. M. Hauser, "The Melting of Finite Steel Slabs in Flowing Nuclear Reactor Fuel," *Nucl. Eng. Des.*, vol. 52, p. 411-428, 1979.

14-46. J. J. Sienicki and B. W. Spencer, Unpublished information, Argonne National Laboratory, 1981.

14-47. M. W. Maresca and O. E. Dwyer, "Heat Transfer to Mercury Flowing In-line Through a Bundle of Circular Rods," *Trans. ASME, J. Heat Transfer*, vol. 86, p. 180-186, May 1964.

14-48. John G. Collier, *Convective Boiling and Condensation*, 2nd. Ed., McGraw-Hill Book Company, New York, p. 33, 1981.

14-49. G. B. Wallis, *One-Dimensional Two-Phase Flow*, McGraw-Hill Book Company, New York, 1969.

14-50. M. M. El-Wakil, *Nuclear Power Engineering*, McGraw-Hill Book Company, New York, 1962.

14-51. Kalimullah and H. U. Wider, Unpublished information, Argonne National Laboratory, 1983.

14-52. G. L. Bordner, et al., Unpublished information, Argonne National Laboratory, 1978.

14-53. A. K. Fisher, et al., "Fuel Dynamics Loss-of-Flow Test L3," ANL-76-79, Argonne National Laboratory, Illinois, June 1976.

14-54. J. G. Eberhart, et al., Unpublished information, Argonne National Laboratory, 1976.

14-55. Argonne National Laboratory, Unpublished information, 1980.

14-56. R. Courant and K. O. Friedrichs, *Supersonic Flow and Shock Waves*, Interscience Publishers, Inc., New York, 1948.

14-57. Argonne National Laboratory, Unpublished information, Argonne National Laboratory, 1977.

14-58. M. Ishii, Argonne National Laboratory, personal communication, 1980.

14-59. Argonne National Laboratory, Unpublished information, Argonne National Laboratory, 1976.

14-60. J. K. Fink, M. G. Chasanov, and L. Leibowitz, Unpublished information, Argonne National Laboratory, 1982.

14-61. G. H. Golden and J. V. Tokar, "Thermophysical Properties of Sodium," ANL-7323, Argonne National Laboratory, Illinois, August 1967.

14-62. J. O. Hirschfelder, C. F. Curtiss, and R. B. Bird, *Molecular Theory of Gases and Liquids*, John Wiley and Sons, Inc., New York, 1954.

14-63. Kalimullah and A. M. Tentner, Unpublished information, Argonne National Laboratory, 1984.

14.10 Nomenclature

Symbol	Definition	Units
l	Linear	
n	Time step n	
‘	Prime always indicates that a quantity refers to a unit of generalized smear volume	
Subscripts		
ac	Acoustic	
an	Annular	
bb	Bubbly or bubble	
bk	Lower boundary of node K	
ca	Cavity	
ch	Coolant channel	
cl	Cladding	
co	Condensation	
deet	De-entrainment	
drag	Drag	
ej	Ejected or ejection	
et	Entrained or entrainment	
fc	Fuel/clad interface	
ff	Frozen fuel	
ffcl	Frozen fuel on cladding	

continues on next page

Table 14.10.1 – continued from previous page

Symbol	Definition	Units
ffsr	Frozen fuel on structure	
fi	Free fission gas in coolant channel	
fica	Free fission gas in molten pin cavity	
ficaxx	Used for special fission gas volume fraction calculation (see Eq. (14.2-46))	
fm	Film (stationary liquid sodium film or moving annular fuel film)	
fr	Friction	
friction	Friction	
FR1	Flow regime 1 - particulate fuel flow	
FR3	Flow regime 3 - partially or fully annular flow	
FR4	Flow regime 4 - bubbly fuel flow	
fsca	Dissolved fission gas in the molten pin cavity	
fs	Dissolved fission gas in the coolant channel	
fu	Liquid and/or mobile solid fuel in the coolant channel	
fuca	Mobile fuel in the molten pin cavity	
fuch	Mobile fuel in the coolant channel (including fuel vapor)	
fufi	Fuel and fission gas	
fv	Fuel vapor in the coolant channel	
fvca	Fuel vapor in the molten pin cavity	
fz	Freezing, frozen	
hoop	Hoop (stress)	
Ht	Heat transfer	
i	Axial coolant channel index, specific enthalpy	
IB	Uppermost slug segment of lower sodium slug	
if	Interface	
in	Inner (middle) clad node or structural node facing neighboring hexan wall	
inlet	Subassembly inlet	
IT	Lowermost slug segment of upper sodium slug	
K	Axial pin or cavity index	
k	Material component k; this can also be cladding or structure	
l	Dummy index	
liq	Liquidus	
ls	Lower slug	
m	m = 1 indicates upward flow, m = 2 downward flow	
me	Melting, melt-in	
mi	Mixture of sodium and fission gas	
min	Minimum	
mv	Moving	
mx	Maximum	
Na	Sodium	
Nl	Liquid sodium	
Nm	Moving sodium liquid and sodium vapor	
node	Node	
Nv	Sodium vapor	

continues on next page

Table 14.10.1 – continued from previous page

Symbol	Definition	Units
op	Open channel - channel volume or cross section remaining open after plateout has occurred	
or	Orifice	
os	Outer surface of clad or structure of fuel film or fuel crust. All these surfaces are facing the coolant channel.	
outlet	Subassembly outlet	
p	Subscript of specific heat for constant pressure	
Pa	Fuel particle	
pin,pi	Pin	
PL	PLUTO2	
power	Power	
radial	Radial	
rl	Dissolved fission-gas release	
S'	Sink or source per unit of generalized smear volume	
sol	Solidus	
sonic	Sonic	
sr	Structure	
un	Uncompressed	
us	Upper liquid sodium slug	
UTS	Ultimate tensile strength	
vg	Vapor and gas	
vi	Viscous	
x	Auxiliary subscript	
z	Axial location z	
zi	Designates bottom of a liquid sodium slug segment	
Greek Symbols	**Definition**	**Units**
α	Void fraction	
β	Delayed neutron fraction	
γ	C_p/C_v	
δ	Partial derivative	
Δ	Derivative in finite differences	
κ	Compressibility	
π	Circumference of a circle divided by its diameter	
ρ	Theoretical density	
ρ'	Generalize linear density	
σ	Surface tension	
θ	Generalized volume fraction	
Regular Symbols	**Description**	**Units**
A	Cross section area	
A'	Interaction or surface area per unit of generalized smear volume	
AFR	Input, liquid sodium friction coefficient	
AFRV	Input, sodium vapor friction coefficient	
AHELP	Auxiliary quantity in momentum conservation	
AMIIN	Auxiliary quantity in momentum conservation	
ARCH	Cross section area of coolant channel per pin	

continues on next page

Table 14.10.1 – continued from previous page

Symbol	Definition	Units
ARCL	Cladding surface area per unit of generalized smear volume	
AREA	Area	
ARFF	Cross section area of plated-out fuel pin	
ARMF	Cross section area of moving fuel per pin	
ARSR	Structure surface area per unit of generalized smear volume	
AXMX	Input, reference cross section area; recommended input value is subassembly cross section area	
BFR	Input, exponent of liquid friction coefficient	
BFRV	Input, exponent of vapor friction coefficient	
BHELP	Auxiliary quantity in the momentum equation	
BMIIN	Auxiliary quantity in the momentum equation	
C	Specific heat	
CDFU	Input, fuel conductivity	
CDNL	Input, liquid sodium conductivity	
CDVG	Input, conductivity of sodium vapor	
CFCOFV	Input, fuel vapor condensation coefficient	
CFFFCL	Fraction of channel perimeter covered by plated-out fuel	
CFFRMF	Friction coefficient for annular fuel	
CFFUCL	Fraction of cladding and structure covered by moving or plated-out fuel	
CFMELT	Fraction of frozen fuel remelting per PLUTO2 time step	
CFMFCL	Fraction of channel perimeter covered by moving fuel	
CFMFFF	Fraction of plated-out fuel covered by moving fuel	
CFNACL	Fraction of channel perimeter that is in contact with sodium	
CFNACN	Input, sodium vapor condensation coefficient	$J/(m^2 \cdot s \cdot K)$
CFNAEV	Input, sodium evaporation coefficient	$J/(m^2 \cdot s \cdot K)$
CIANIN	Input, determines when a complete annular fuel flow begins	
CIA1	Input, constant in the fuel particle-to-sodium heat transfer	
CIA2	Input, constant in the fuel particle-to-sodium heat transfer	
CIA3	Input, constant in the Deissler heat-transfer correlation	
CIA4	Input, controls range of interpolation between boiling heat transfer coefficient and single-phase gas heat-transfer coefficient	
CIA5	Input, controls drag dependence in void fraction in the particulate flow regime	
CIA6	Input, controls drag in the bubbly flow regime	
CIBBIN	Input, controlling onset of bubbly flow regime	
CIETFU	Input, controls effect of fuel particles on sodium film entrainment	
CIFRFU	Input, part of the fuel friction calculation	
CIFUFZ	Input, controls mode of fuel freezing	

continues on next page

Table 14.10.1 – continued from previous page

Symbol	Definition	Units
CIFUMO	Input, controls axial fuel momentum low on ejection from pin	
CINAFO	Input, initial and maximum sodium film fraction	
CIREFU	Input, Reynolds number above which fully turbulent fuel flow is assumed for friction calculation	
CIRTFS	Input, controls dissolved gas release	1/s
CIVOID	Input, controls liquid sodium fraction below which single-phase heat-transfer and friction laws are used	
CMFU	Input, liquid fuel compressibility	1/Pa
CMNL	Input, liquid sodium compressibility	1/Pa
COFICH	Convective fission-gas mass flux in coolant channel	$kg/(s \cdot m^2)$
COFUCH	Convective fuel mass flux in coolant channel	$kg/(s \cdot m^2)$
COFUOS	Convective fuel energy flux in the channel	$J/(s \cdot m^2)$
CONACH	Convective sodium mass flux in coolant channel	$kg/(s \cdot m^2)$
CPFU	Input, fuel heat capacity	$J/(s \cdot m^2)$
CTFRFU	Fraction of the moving fuel that is in contact with clad or plated-out fuel	
C1	Input, part of liquid sodium heat-transfer correlation	
C2	Input, part of liquid sodium heat-transfer correlation	
C3	Input, part of liquid sodium heat-transfer correlation	
C1VIPR	Input, used for the artificial viscous pressure calculation in the pin	
C2VIPR	Input, used for the artificial viscous pressure calculation in the pin	
D	Diameter, hydraulic diameter	m
DEFICH	Generalized smear density of free fission gas in the coolant channel	kg/m^3
DEFUCH	Generalized smear density of the moving fuel in the coolant channel	kg/m^3
DEFVCH	Generalized smear density of fuel vapor in the coolant channel	kg/m^3
DENACH	Generalized smear density of sodium in the coolant channel	kg/m^3
DENMCH	Generalized smear density of moving sodium in the coolant channel	kg/m^3
DZPLIN	Input, minimum length of a Lagrangian edge cell	m
e	Internal energy	J/kg
EGBBLY	Input, internal fuel energy below which fuel freezing begins	J/kg
EGFULQ	Input, internal fuel energy at the solidus	J/kg
EGFUSO	Input, internal fuel energy at the solidus	J/kg
EGMN	Input, internal fuel energy below which fuel cannot switch from the particulate to a continuous flow regime	J/kg
EPCH	Input, controls use of advanced pressure in in-pin calculation	
f	Factor used in interphase drag term	
F	Friction factor	
FF	Mass of fuel ejected per time step and per unit of generalized smear volume	kg/m^2
FIFNGB	Input, fraction of fission gas on grain boundaries which becomes available as free gas upon fuel melting	

continues on next page

Table 14.10.1 – continued from previous page

Symbol	Definition	Units
FNARME	Input, cross section area of pin cavity/cross section area of fuel above which failure of this node is allowed to occur	
FNFUAN	Input, controls when annular fuel flow in molten pin cavity is assumed	
FNMECA	Fraction of node width of radial heat-transfer node on cavity boundaries which melts in pure PLUTO2 time step	
FNMELT	Input, controls locus of cavity boundary	
FNPI	Fraction of all the pins in a subassembly which have failed	
FNPOHE	Relative power level	
FN	Fraction of the molten fuel in a ejection cell which is ejected per PLUTO2 time step	
FUELMS	Initial fuel mass in a radial fuel-pin mode	kg
FUMASS	Initial total fuel mass in an axial fuel node	kg
FUMS	Current fuel mass in the radial fuel pin node on the cavity boundary	kg
F.R. 1	Particulate fuel flow regime	
F.R. 3	Partially or fully annular flow regime	
F.R. 4	Bubbly fuel flow regime	
g	Gravity	m/s^2
GAMSS	Input, fraction of power going into direct heating of structure	
GAMTNC	Input, fraction of power going into direct heating of coolant	
GAMTNE	Input, fraction of power going into direct heating of cladding	
h	Heat-transfer coefficient	$J/(m^2 \cdot s \cdot K)$
H	Heat-transfer coefficient times heat-transfer area	$J/(s \cdot K)$
HCCLMI	Input, heat-transfer coefficient between hot cladding and a two-phase sodium/gas mixture	$J/(m^2 \cdot s \cdot K)$
HCFFMI	Input, heat-transfer coefficient between frozen fuel crust surface and two-phase sodium/gas mixture	$J/(m^2 \cdot s \cdot K)$
HCFUBB	Input, heat-transfer coefficient between the interior of the molten fuel and bubble surface	$J/(m^2 \cdot s \cdot K)$
h1	Auxiliary heat-transfer coefficient	$J/(m^2 \cdot s \cdot K)$
h2	Auxiliary heat-transfer coefficients	$J/(m^2 \cdot s \cdot K)$
I	Axial node index in coolant channel	
IB	Channel zone in which the uppermost segment of the lower slug is located	
IDIFF	Offset between pin and channel grid. The first pin node is at the same elevation as channel node IDIFF + 1	
IFLAG	Pointer array which gives the flow regime number for each axial channel node	
IMAX	Uppermost slug segment of upper sodium slug	

continues on next page

Table 14.10.1 – continued from previous page

Symbol	Definition	Units
IT	Channel zone in which the lowermost segment of the upper slug is located	
k	Conductivity	J/(m·s·K)
K	Axial index in the pin	
KKMX	Uppermost node in the molten pin cavity	
KZPIN	Coolant channel zone which contains the fuel pins	
KK1	Lowermost node in the molten pin cavity	
L	Length	m
L1	Length of the Lagrangian node at one end of the interaction region	m
L2	Length of Lagrangian node at the other end of the interaction region	m
N’	Number per unit of generalized smear volume	
NCPLEV	Input, number of cladding nodes which have to be molten in to lead to switch to LEVITATE	
NGRDSP	Input, number of grid spacers in the channel zone containing fuel pins	
NRPI	Number of pins per subassembly	
NT	Number of radial fuel pin nodes	
Nu	Nusselt number	
P	Pressure	Pa
PECH	Perimeter of the coolant channel associated with one pin	M
PORFR	Porosity fraction	
POW	Input, power in highest rated axial fuel pin node	W
POWCOF	Exponent in the exponential function which give the power history during main time step	
PRFAIL	Input, failure pressure for non-mechanistic clad failure Propagation	Pa
Pr	Prandtl number	
PSHAPE	Input, axial pin power shape	
PSHAPR	Input, radial power shape in a pin	
Q	Fission heat source	W/kg
r	Radius	M
R	Gas constant or radius	J/(kg·K)
RAFPLA	Input, radius of large particles	m
RAFPSM	Input, radius of small particles (which have been generated due to the decay of the larger particles after TIFP seconds)	kg
RETFG2	Retained fission-gas mass in the original radial fuel-pin node at the cavity boundary before it began melting into the cavity	
Re	Reynold’s number	
RGAS	Input, gas constant for fission gas	J/(kg·K)
RGNA	Sodium vapor gas constant which is a calculated quantity in PLUTO2	J/(kg·K)
S	Mass sink or source	kg/s

continues on next page

Table 14.10.1 – continued from previous page

Symbol	Definition	Units
S’	Mass sink or source per unit of generalized smear volume	kg/(s·m^3)
St	Stanton number	
t	Time	s
T	Temperature	K
TECLMN	Input, cladding temperature above which frozen fuel will not stick to the cladding	K
TECLRL	Input, cladding temperature above which the frozen fuel crust will be released	K
TEFAIL	Input, cladding failure temperature in nonmechanistic failure criteria	K
TESOL	Input, steel solidus temperature	K
TIFP	Input, time delay for fragmentation of large particles into small ones	s
TKFF	Thickness of frozen fuel crust	m
u	Velocity	m/s
UFCH	Fuel velocity in the channel	m/s
UMCH	Sodium/gas mixture velocity in the channel	m/s
VCONST	Auxiliary quantity in the sodium film entrainment calculation	
VFNALQ	Input, liquid sodium fraction below which a particulate flow regime can become a continuous one	
VFNARE	Input, liquid sodium fraction below which a particular flow regime can become a continuous one	
VIFG	Input, viscosity of sodium vapor	kg/(m·s)
VIFULQ	Input, viscosity of liquid sodium	kg/(m·s)
VOLUME	Volume	m^3
w	Width	m
W	Mass flow rate	kg/s
x	Sodium quality	
XKORGD	Input, orifice coefficient of a single grid spacer	
XKORV	Input, expansion, contraction, or orifice coefficient	
z	Axial coordinate	m

CHAPTER

FIFTEEN

PINACLE: IN-PIN PRE-FAILURE MOLTEN FUEL RELOCATION MODULE

15.1 Overview

15.1.1 Historical Background

During both the LOF and TOP postulated accidents, the mismatch between the energy generated in the fuel pin and the energy removed by the coolant may lead to the overheating of the fuel pin. During the early period, limited fuel relocation occurs due to the axial expansion of the solid fuel pin, which may reduce or increase the core reactivity, depending on the power response. As the accident proceeds, the inside of the fuel pin begins to melt, leading to the formation of an internal cavity as shown in Figure 15.1.1.

This cavity is filled with a mixture of molten fuel and fission gas and expands continuously, both radially and axially, due to continued fuel melting. The fuel-gas mixture in the cavity is pressurized due to the presence of fission gas and can move under the influence of the local pressure gradients. During this period fuel relocation occurs due to both axial extrusion of the solid fuel pin and the in-pin hydrodynamic relocation of the molten fuel. As long as the cavity maintains a bottled-up configuration the hydrodynamic fuel relocation is limited and tends to introduce an amount of negative reactivity comparable in magnitude to the negative reactivity introduced by axial extrusion of the solid fuel. As the cavity walls continue to melt there is a competition between the two effects illustrated in Figure 15.1.2:

c) The radial extension of the cavity and cladding melting which can cause fuel pin failure. When pin failure occurs, the inner cavity is connected to the coolant channel, which is at a significantly lower pressure, and the molten fuel inside the pin is accelerated rapidly toward the pin failure location. This initial in-pin fuel relocation can have either a negative or positive reactivity contribution, depending on the failure location and axial failure propagation. Molten fuel is ejected into the coolant channel where it is dispersed axially. This fuel dispersal leads to a large insertion of negative reactivity and eventual neutronic shutdown of the reactor.

d) The axial extension of the cavity, which can cause the cavity to reach the top of the fuel pin. When this happens the pressurized molten fuel in the cavity is connected to the lower pressure upper plenum and can relocate suddenly, leading to a large insertion of negative reactivity and possible neutronic shutdown of the reactor.

The traditional pressure relief mechanism in the SAS4A [15-1] context was the rupture of fuel pin cladding leading to the onset of fuel motion. While the post pin-failure fuel relocation was modeled in considerable detail by the LEVITATE [15-2] and PLUTO [15-3] models, the in-pin relocation of molten fuel prior to pin failure was not modeled.

The new PINACLE [15-4] code that has been implemented in SAS4A provides the capability to model the dynamic relocation of the in-pin molten fuel prior to cladding failure. PINACLE is an Eulerian, two-phase,

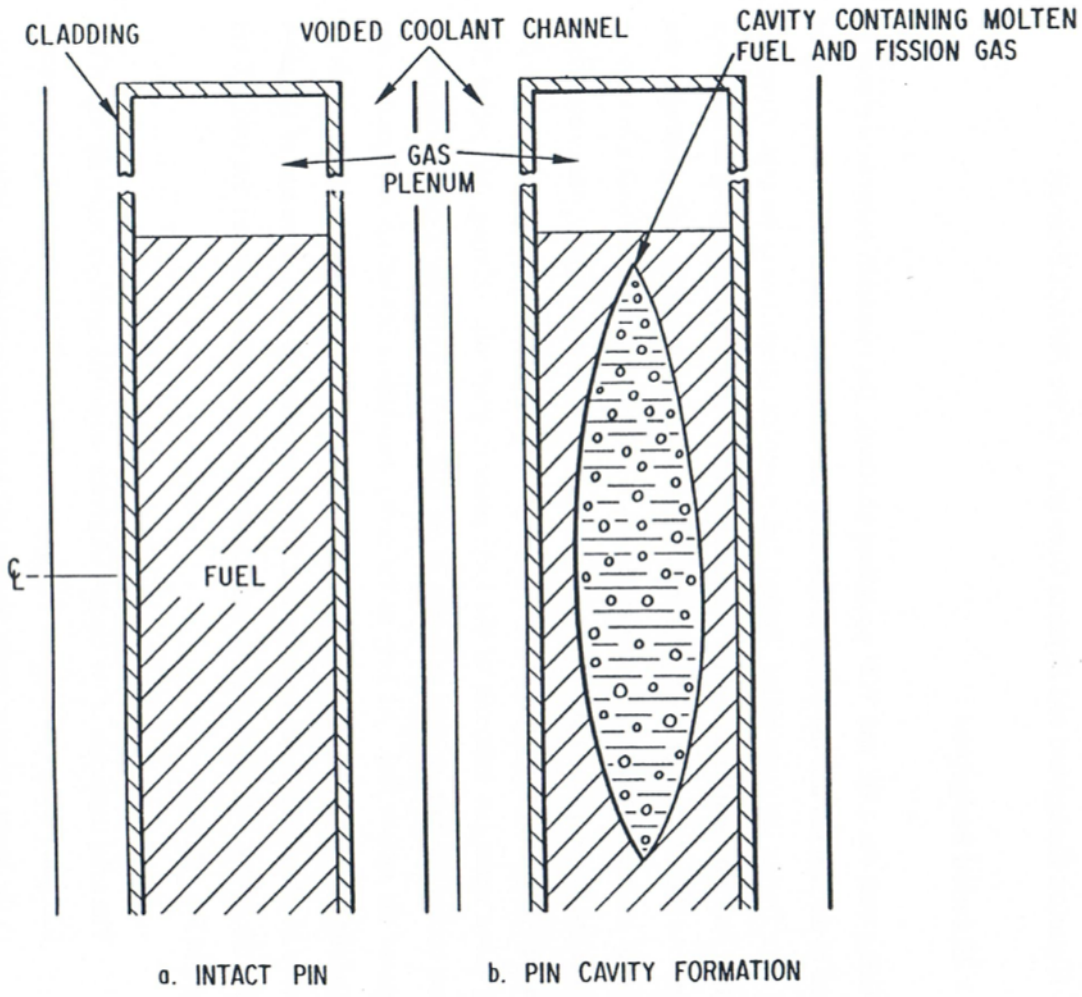


Figure 15.1.1: Cavity Formation During Initial Accident Phase

transient hydrodynamic model describing the axial fuel relocation in a variable area geometry. It has been constructed using the same computational variables and method of solution as LEVITATE and PLUTO. The compatibility of PINACLE with these two models allows SAS4A to provide a consistent treatment of the in-pin fuel relocation from melting to the end of the initiating phase.

The components tracked by PINACLE are the molten fuel and two types of fission gas. The fission gas can exist either in the form of small bubbles, constrained by surface tension, which do not contribute significantly to the cavity pressure or as free gas which pressurizes the surrounding molten fuel. The small bubbles coalesce in time and gradually become part of the free gas field.

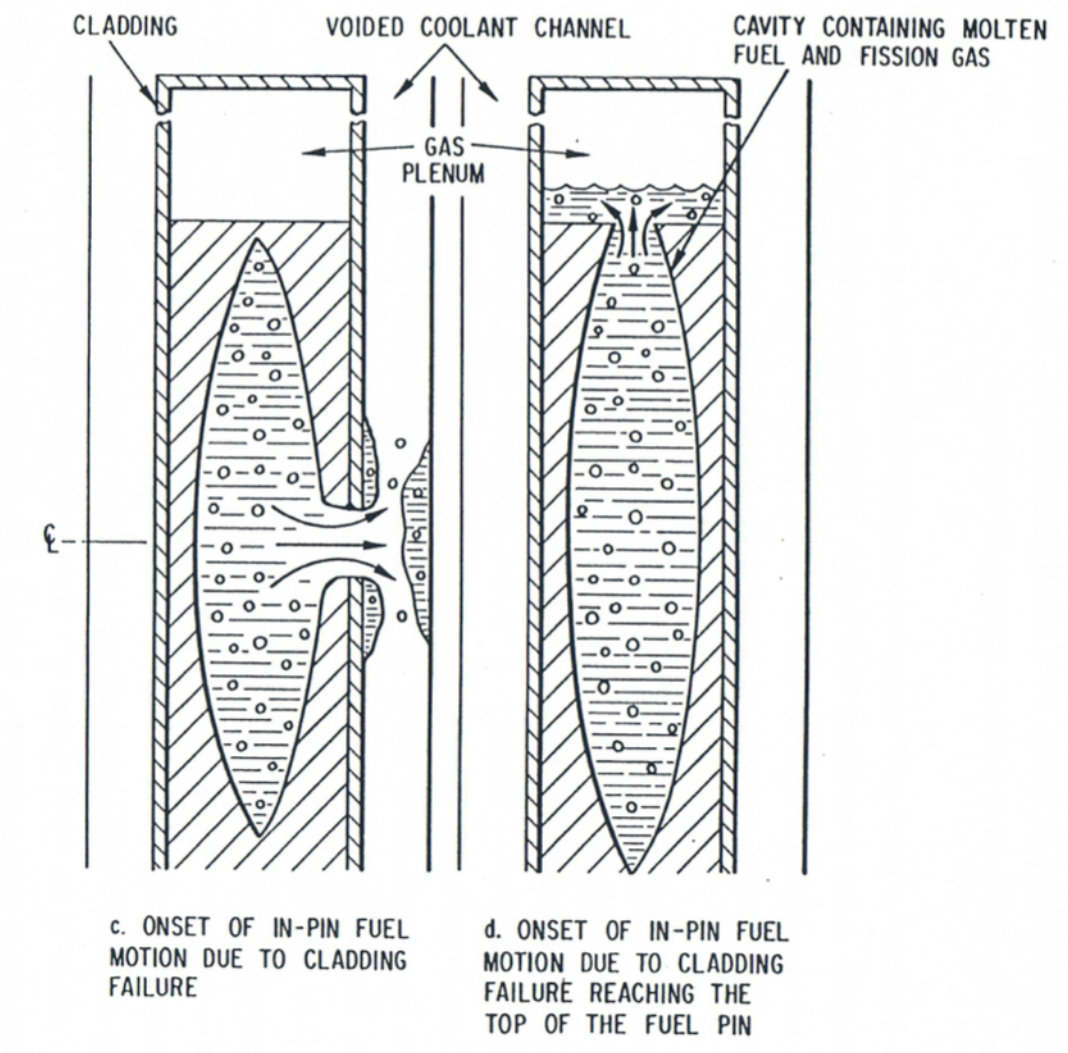


Figure 15.1.2: Molten Fuel Relocation Modes

To advance the numerical solution, PINACLE uses a staggered mesh, with the dependent variables, density and enthalpy, defined at the center of each cell, and the velocities defined at the cell boundaries. Only a bubbly flow regime is currently modeled, with the assumption that the molten fuel and the fission gas are well mixed and move with the same velocity at a given axial location.

Pre-failure in-pin fuel motion can play a particularly significant role in metal fuel cores and in oxide fuel cores subjected to a slow ramp transient overpower (TOP) excursion. In these cases the molten fuel cavity can extend

all the way to the top of the pin and allow significant in-pin molten fuel relocation prior to cladding failure. The ejection of the molten fuel into the gas plenum space can provide an important source of negative reactivity.

15.1.2 Physical Description of the PINACLE Model

15.1.2.1 Cavity Formation and In-pin Fuel Motion

As the accident proceeds, the inside of the fuel pin begins to melt, leading to the formation of an internal cavity. This cavity is filled with a mixture of molten fuel and fission gas, and expands continuously, both radially and axially, due to fuel melting. The fuel-gas mixture in the cavity is pressurized due to the presence of fission gas and fuel vapor. A limited amount of axial fuel relocation occurs in this bottled-up configuration. PINACLE models this molten fuel relocation, the heat transfer between the molten fuel in the cavity and the solid fuel walls, and the cavity extension. Note that while in oxide fuel pins and in metal U-Fissium pins the internal molten cavity is likely to be centrally located within the pin at any axial location, the situation is different for the U-Pu-Zr fuel pins. Due to the Zr migration and the formation of a middle Zr-depleted region, the molten cavity in these pins is likely to develop as an annular cavity. The current PINACLE version only treats circular cavities, but the modeling of the annular cavity formation will be added in a future version.

The hydrodynamic calculations in the coolant channel and the heat transfer in the solid fuel are not modeled by PINACLE. These calculations are performed by other SAS4A modules as described in Section 15.1.3.

15.1.2.1.1 In-pin Molten Fuel Relocation

In metal fuel cores and in oxide cores subjected to slow ramp TOP's it is possible that the molten cavity will reach the top of the active fuel column prior to the cladding failure. If no upper blanket pellets are present, as was the case in the metal fuel pins used in the M2 and M3 experiment [15-9], the pressurized molten fuel in the cavity is offered an escape path to the lower pressure upper gas plenum. As it escapes from the pressurized cavity, the fuel will displace the liquid sodium slug present above the top of the pin and will reduce the free volume available in the gas plenum. The pressure in the cavity will decrease as the molten fuel is ejected into the plenum, while the pressure in the plenum will increase due to the volume reduction. After the initial fuel burst, a quasi-equilibrium will be established, with more fuel being ejected, at a slower rate, as the fuel melting in the cavity continues.

If blanket pellets are present above the active fuel column, PINACLE assumes that these pellets can move freely upward, for a distance FUSDLT. The input value FUSLDT is defined by the actual pin construction. Thus, in pins with dimples limiting the axial fuel motion, the input FUSLDT is determined by the distance between the top of the blanket stack and the dimples. When the fuel is ejected above the active fuel column it displaces the blanket pellets and creates a fuel-filled space above the active fuel. When the blanket stack reaches the dimples, or another rigid obstacle, its motion stops and the axial fuel ejection is reduced significantly. The ejection can still continue at a low rate by the ejection of limited amounts of fuel into the space already available above the active fuel column.

15.1.2.2 Geometry Description

The geometry modeled in PINACLE and its relationship to the fuel assembly modelled by SAS4A is shown in Figure 15.1.3.

PINACLE calculates all the thermal hydraulic events that occur in the fuel pin cavity. Although only one pin is shown in Figure 15.1.3. PINACLE will account for the appropriate number of pins per subassembly, as specified in the input description. The hydrodynamic in-pin calculations are performed on a mesh grid using the subscript K, having the origin at the bottom of the lower blanket. The top node of the upper blanket is indicated by the variable MZ. The active fuel core extends from KCORE1 and KCORE2. The fuel pin cavity, which increases gradually both radially and axially, extends form KK1 to KKMX. It cannot extend beyond the active core boundaries.

The upper blanket or liquid sodium above the active fuel column extends initially from KCORE2+1 to MZ. When the molten cavity reaches the top of the fuel column the molten fuel in the cavity is ejected above the fuel column, displacing the blanket pellets and/or liquid sodium, as shown in Figure 15.1.3. The initial fuel ejection is determined by the pressure difference between the upper gas plenum and the top cavity cell. When the upper blanket stack reaches a rigid obstacle, fuel ejection is determined by the pressure difference between the space above the active fuel and the top cavity node.

15.1.3 Interaction of PINACLE with other Models Within the SAS4A System

The PINACLE model is fully integrated within the SAS4A whole core accident analysis code. PINACLE is initiated when the accident sequence, as modeled by other modules, leads to the internal melting of the fuel pins in some of the subassemblies. During its calculations PINACLE exchanges information with other SAS4A models such as DEFORM-4 or the point kinetics model, which describe other phenomena affecting the same computational channel. Finally when the cladding failure occurs in a computational channel PINACLE will transfer control to other models such as PLUTO2 or LEVITATE, which will continue the calculations in the channel.

This chapter describes the important aspects related to the initiation, calculations and termination of PINACLE execution. A simplified modular representation of the relationships between PINACLE and other SAS4A modules is presented in Figure 15.1.4.

15.1.3.1 PINACLE Initiation

PINACLE can be initiated only by one routine of the SAS4A code, i.e., CAVMOT.

The initiation of the in-pin molten fuel motion is decided, for any given channel, in the routine CAVMOT, called from DFORM3. CAVMOT checks if the maximum areal melt fraction (defined by FNMELT times the heat of fusion) in the pin has reached the input value **FPINAC**. If this condition has been met, CAVMOT checks if there are at least 3 adjacent axial segments with a melt fraction higher than FPINAC*CPINAC, where **CPINAC** is an input constant. If this condition has also been met, CAVMOT will set the flag IPINAC=1 and will begin the PINACLE initiation by calling the PNINIT routine. It should be noted here that the routine CAVMOT is called even if the DEFORM pin mechanics model is not used i.e., when ISSFU2=0.

The PNINIT routine prepares a number of variable necessary for PINACLE and then calls two initialization routines, PNINPT and PNSET. These routines complete the PINACLE initialization. Control is then returned to TSTHRM, via PNINIT, CAVMOT and DFORM3.

It is important to note that, after the PINACLE initiation, the coolant channel hydrodynamic calculation is performed by the boiling module, even if only subcooled sodium is present in the channel. This is a temporary situation, due to the fact that the HTRVPN heat transfer module developed for PINACLE can only interface with the boiling model.

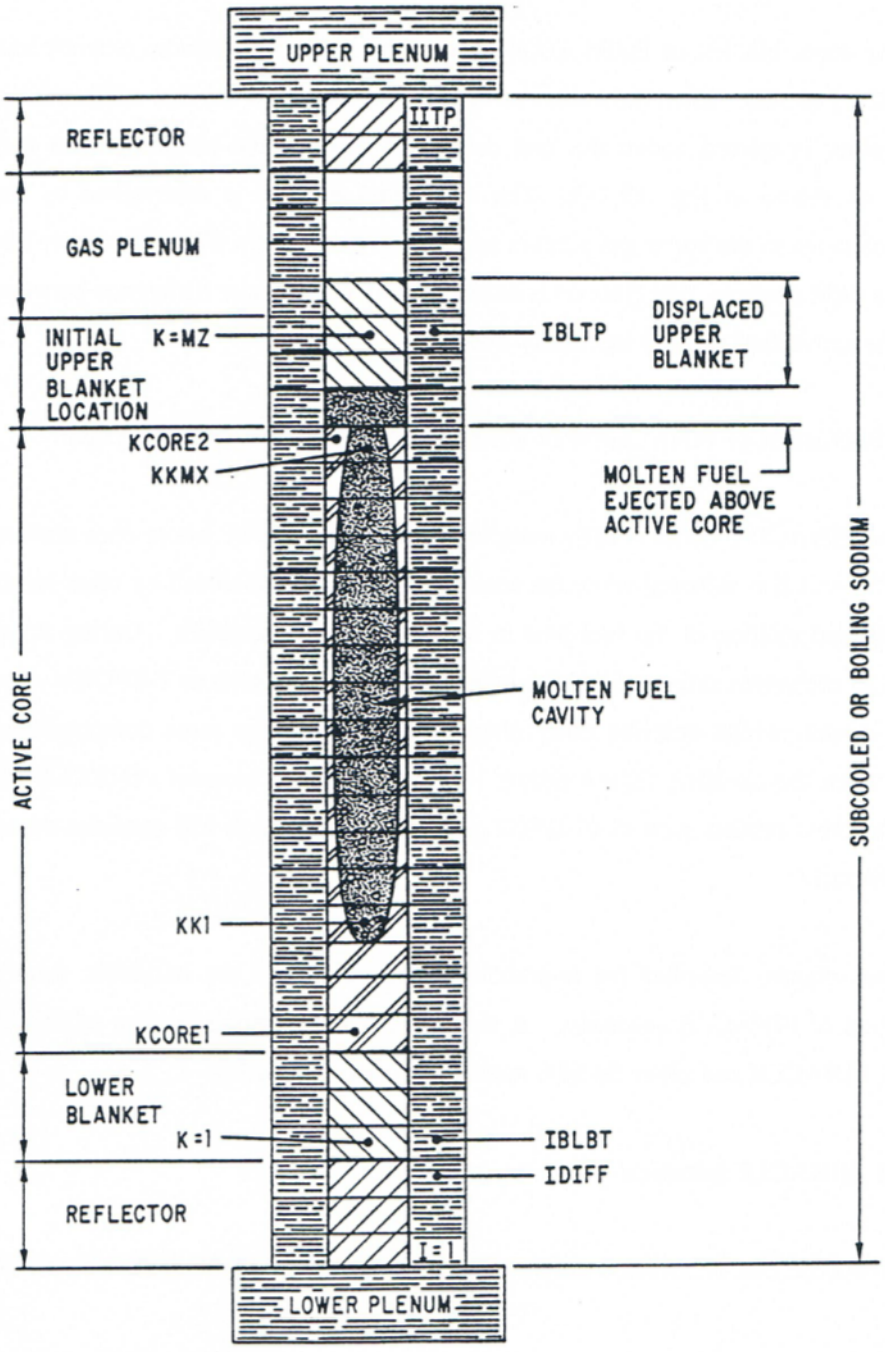


Figure 15.1.3: Geometry Modeled by PINACLE

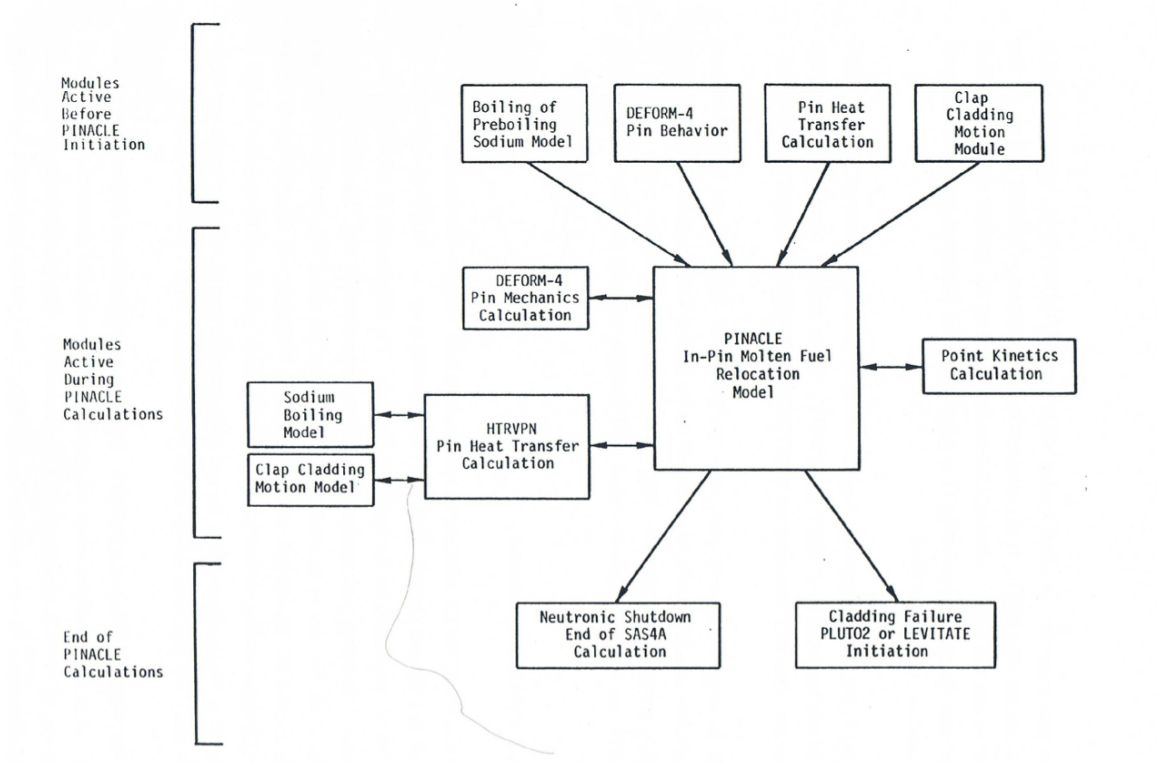


Figure 15.1.4: Relationship Between PINACLE and Other SAS4A Modules

15.1.3.2 PINACLE Calculations

Once the PINACLE initialization routines have been executed and the flag IPINAC has been set to 1 for that channel, ICH, the SAS4A transient driver, TSTHRM, will begin the execution of calculations for the remaining channels. The initialization of PINACLE in any given channel is always performed at the end of a heat transfer time step. The time is advanced then for all the channels by the coolant time step. When the end of the next heat transfer time step is reached for the channel where IPINAC=1, TSTHRM will transfer control to the PINACLE driver routine, PINACL. PINACL will retain control and advance the solution in the pin cavity for the channel from the end of the previous heat transfer step to the end of the current one. The flow chart in Figure 15.1.5 shows the logic of the PINACLE driver.

First PINACL will execute PNSET2. This subroutine initializes all temporary integers and arrays. These are values that can be calculated using the permanent quantities. They are kept only as long as PINACL retains control in the channel ICH. The solution for the hydrodynamic in-pin fuel motion is then advanced for the channel by calling the routine PN1PIN and PN2PIN.

Next, the PINACL driver routine determines the maximum time step acceptable for the hydrodynamic calculation in the next cycle. It is noted that the PINACLE time step is not allowed to exceed the time remaining until the end of the heat-transfer time step. If this happens, the PINACLE time step will be cut back, so that the end of the next PINACLE time step will coincide with the end of the heat transfer time step. The next task of PINACLE is to calculate the fuel reactivity. This calculation is described in more detail in the section on PINACLE interaction with the FEEDBK routine, *15.1.3.3.3*.

If the end of the PINACLE time step coincides with the end of a heat-transfer time step, the HTRVPN routine is called. This routine calculates the new temperatures in the solid fuel pin and in the cladding at the end of the current heat transfer time step.

If it is time to produce output, PINACL will print the output described in Section 15.3.3. Then PINACL will return control to TSTHRM. When the pin mechanics model DEFORM-4 is used the PINACLE calculation is always followed immediately by the DEFORM-4 calculation.

15.1.3.3 PINACLE Interfaces

15.1.3.3.1 PINACLE Interface with the DEFORM-4 Pin Mechanics Model

PINACLE is tightly coupled to the DEFORM-4 pin mechanics model. After PINACLE initiation in a channel both PINACLE and DEFORM-4 are called successively at the end of each heat transfer time step. PINACLE receives from DEFORM-4 the modified cavity radii via the R(I,J) array and the modified axial length of each axial cell via the ZCOOL(I) array. It then calculates the temperature and composition changes in each cavity cell and, using the modified volume, calculates the new pressure in each cavity cell. These pressures are then passed back to DEFORM-4 via the CAVPRS(J) array and will be used as a boundary condition in the DEFORM calculations during the next heat transfer time step. PINACLE also calculates the fuel melt-in at the cavity boundary and modifies the cavity radius, which is then passed back to DEFORM, via the R(I,J) array. Finally, the axial cavity extension is calculated in the HTRVPN heat transfer module and the new cavity axial boundaries are passed to DEFORM from PINACLE via the KK1DF and KKMXDF integers. Using the information received from PINACLE and the updated pressures in the coolant channel received from the boiling hydrodynamic model, DEFORM calculates the new radial and axial pin dimensions, which are then passed back to PINACLE to be used in the next heat transfer time step calculations.

15.1.3.3.2 PINACLE Interface with the HTRVPN Heat Transfer Model

After the initiation of PINACLE calculations in a particular SAS4A channel, the heat transfer calculations in the solid fuel pin are performed by the HTRVPN heat transfer module. This module is similar to the TSHTRV module used when the boiling model is active but PINACLE has not yet started. HTRVPN differs from TSHTRV in the treatment of the fuel inner boundary condition. While TSHTRV treats a solid or hollow fuel pin with a zero heat flux inner boundary condition, HTRVPN takes into account the presence of the molten fuel cavity, when necessary. If the cavity is present in an axial cell, HTRVPN will perform the heat transfer calculation for the solid fuel only. The heat flux between the molten fuel in the cavity and the inner boundary of the solid fuel is calculated by PINACLE, which integrates the energy transferred each PINACLE time step, from the beginning to the end of the heat transfer time step. The integrated heat flux is then passed to HTRVPN via the HFCAWA(I) array. PINACLE also calculates the fuel melt-in at the cavity boundary and modifies the cavity radius which is made available to HTRVPN via the R(I,J) array and the number of solid radial fuel nodes present at each axial location. The index of the innermost solid fuel node is made available to HTRVPN via the IZJ(I) array. HTRVPN then calculates the new temperatures in the solid fuel, updating the array T2(I,J). These new temperatures will be used by PINACLE in the next heat transfer time step to determine the fuel melt-in and the heat flux at the cavity boundary.

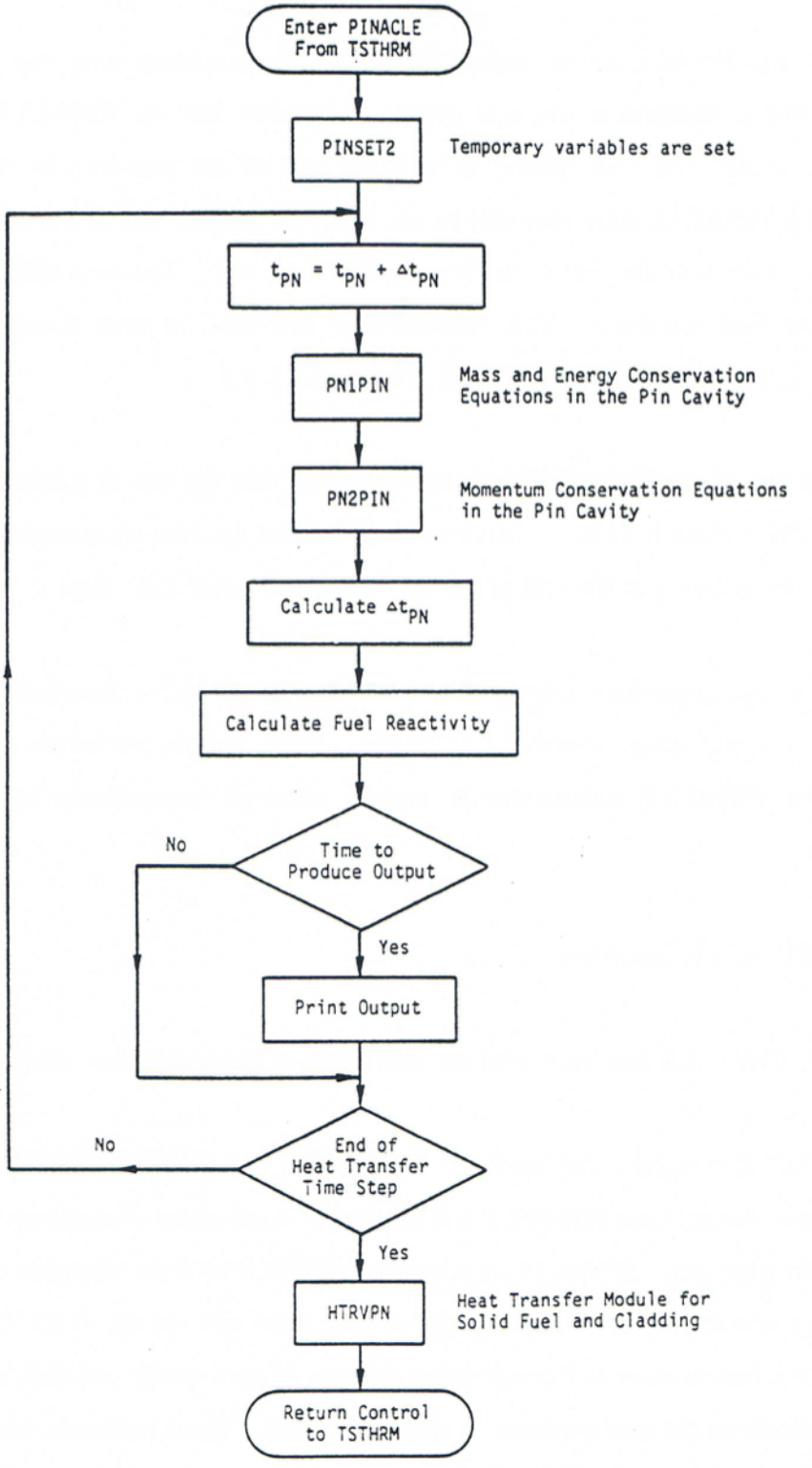


Figure 15.1.5: PINACLE Driver Flowchart

15.1.3.3.3 PINACLE Interface with the FEEDBK Reactivity Module

FEEDBK calculates the data for net reactivity changes for a channel during a primary time step and transfers this information to the neutronic model which calculates the changes in the reactor power. PINACLE calculates the axial fuel mass distribution for the SAS4A channel under its control at the end of each primary time step. In other channels the fuel masses can be updated by other modules, e.g., PLUTO2 or LEVITATE, which have control at the given time. As each channel is calculated, FEEDBK sums the fuel relocation reactivity together with the other reactivity contributions, such as fuel axial expansion, to determine the total reactivity of the core. Using this information, the neutronic model determines the new power level which is used by PINACLE in the following time step. The power level at the end of each PINACLE time step is determined in PINACL, using an exponential fit of the power-time history supplied by the neutronic model. This fit is based on the power level at the beginning of the time step and the precalculated power level at the end of the current main time step. By using this calculated power level and the axial input power distribution, the specific power for each axial segment is calculated.

15.1.3.4 PINACLE Termination

The PINACLE calculation will continue in a channel until the calculation is terminated due to neutronic shutdown (i.e., the total reactivity is less then the input value NFUELD), the fuel freezes in all cavity cells due to lower power levels or the failure routine FAILUR indicates that the cladding failure has occurred. If such a failure occurs, the PINACLE calculation terminates and a transition is made to one of the post-failure fuel motion models, PLUTO2 or LEVITATE. These models will continue to calculate the thermal hydraulic events in the pin cavity in addition to calculating the hydrodynamic events in the coolant channel. Because the PINACLE model was developed using as a starting point the cavity hydrodynamic model used in PLUTO2 and LEVITATE, there is full consistency between the cavity models used in SAS4A before and after the fuel pin failure. However, at present PLUTO2 and LEVITATE do not yet account for the presence of the fuel ejected above the active fuel column which is modeled in PINACLE.

The PINACLE calculations will terminate in a channel where the fuel in the cavity refreezes due to lower power levels. PINACLE will then remain in a stand-by state and could be restarted again in that channel if the conditions require it. This restart feature of PINACLE is not implemented at this time, but will be part of the near-term development effort.

15.2 In-pin Hydrodynamic Model

15.2.1 Overview of the Numerical Approach for the In-pin Fuel Motion Calculation and Description of Subroutines PN1PIN and PN2PIN

There are several requirements for the solution algorithm for this one-dimensional, compressible flow problem with variable flow cross section: (a) it has to be able to handle large mass sources (due to fuel melting at the cavity boundary), (b) it has to conserve mass perfectly, and (c) it has to run efficiently. This was achieved with a predominantly explicit Eulerian solution method. All convective mass, energy, and momentum fluxes are treated explicitly (i.e., the beginning of time-step values are used). However, the solution sequence of the different equations introduces a certain implicitness, which can be important when treating the strong mass sources.

The set of conservation equations describing the in-pin fuel motion includes the following; 3 for mass, 1 for momentum, and 1 for energy. The mass conservation equations are for the fuel, and the free and dissolved fission gas. The fuel and free fission-gas mass conservation equations are solved first, followed by the fuel energy conservation equation. The fission gas temperature change is assumed to be the same as the fuel temperature

change. From the results of the mass and energy conservation equations, a new pressure is calculated. This is not the true end-of-time-step pressure because the velocity changes during the time step have not been included. However, it is a proper prediction for the end-of-time-step value in an explicit sense. The new pressures are then used in the fuel/fission-gas momentum equation. There is an input option in PINACLE to use a combination of the new and the old pressures. This option lets the pressure, P, be:

$$P = (1 - \text{EPCH}) \cdot \text{beginning-of-time-step pressure} \\ + \text{EPCH} \cdot \text{advanced pressure.} \tag{15.2-1}$$

The recommended input value is EPCH $= 1$ because the calculation remained stable for longer time steps in test problems involving shock propagation and shock reflection when this input value was used [15-5].

The time-step criterion in this compressible calculation is the sonic Courant condition. This does not require particularly small time steps because the sonic velocity in two-phase mixtures is fairly low for the void fractions encountered in pin blowdown calculations.

A staggered numerical grid is used with the densities, energies, and temperatures defined at the cell centers, and the velocities defined at the cell edges. The spatial differencing uses full donor cell differencing. Although this is not as accurate as higher order differencing, it makes the calculation very stable.

The free fission-gas mass conservation, the fuel mass conservation, and the fuel energy conservation equations are solved in subroutine PN1PIN (**PIN**ACLE **1**st **PIN** ROUTINE). PN1PIN also computes the molten cavity geometry changes due to fuel melt-in. The fuel/fission-gas momentum equation and the dissolved fission-gas mass equation are solved in PN2PIN (**PIN**ACLE **2**nd **PIN** ROUTINE). This routine also calculates the sonic velocities for each node and the maximum hydrodynamic time step (see Section 15.3.1.3).

15.2.2 Definition of the Generalized Smear Densities for the In-pin Hydrodynamic Calculation

The use of generalized smear densities in SAS4A for the PLUTO2 [15-3] and LEVITATE [15-2] fuel motion models was prompted by the many different moving and stationary components treated by these models. PINACLE, which has used the in-pin fuel motion model in PLUTO2 and LEVITATE as a starting point, has maintained the use of the smeared densities for consistency and convenience. The use of this concept also simplified the differential and finite difference equations for variable cross section flow. The pie chart in Figure 15.2.1 gives an example of the relative cross sectional areas within a subassembly or experimental loop at a certain axial elevation

If the total area of the subassembly is AXMX, the generalized volume fraction of a certain component k is:

$$\theta_k(z,t) = \frac{A_{\text{k}}(z,t)}{\text{AXMX}} \tag{15.2-2}$$

where A_{k} is the cross sectional area occupied by component k (the latter refers, e.g., to the cavities or the moving fuel in all failed pins within the subassembly cross section). The reference area AXMX, which is an input quantity, can be arbitrarily chosen. This is because the code is invariant to the choice of AXMX (i.e., as long as it is not varied by several orders of magnitude which can lead to differences due to truncation errors). However, the recommended value of AXMX is the cross sectional area of a subassembly or experiment test section (including the can wall) because the volume fractions of the different components that appear in the PINACLE output are better understood in this case.

The generalized volume fraction of the free fission gas and fuel vapor in the cavity is the difference between the cavity volume fraction and the fuel volume fraction.

$$\theta_{\text{fica}}(z,t) = \theta_{\text{ca}}(z,t) - \theta_{\text{fuca}}(z,t) \tag{15.2-3}$$

where:

θ_{fica} = Generalized volume fraction of the free fission gas and fuel vapor in the pin cavities

θ_{ca} = Generalized volume fraction of the molten cavities in all pins which can be calculated from $A_{\text{ca}}/\text{AXMX}$

θ_{fuca} = Generalized volume fraction of the fuel in the cavities of all pins

Generalized smear densities, which are always marked by a prime, are defined as products of physical densities and generalized volume fractions:

$$\rho'_{\text{fuca}}(z,t) = \rho_{\text{fuca}}(T) \cdot \theta_{\text{fuca}}(z,t) = \rho_{\text{fuca}}(T) \cdot \frac{A_{\text{fuca}}(z,t)}{\text{AXMX}} \tag{15.2-4}$$

$$\rho'_{\text{fica}}(z,t) = \rho_{\text{fica}}(T) \cdot \theta_{\text{fica}}(z,t) = \rho_{\text{fica}}(T) \cdot \frac{A_{\text{fica}}(z,t)}{\text{AXMX}} \tag{15.2-5}$$

$$\rho'_{\text{fsca}}(z,t) = \rho_{\text{fsca}}(T) \cdot \theta_{\text{fuca}}(z,t) = \rho_{\text{fsca}}(T) \cdot \frac{A_{\text{fuca}}(z,t)}{\text{AXMX}} \tag{15.2-6}$$

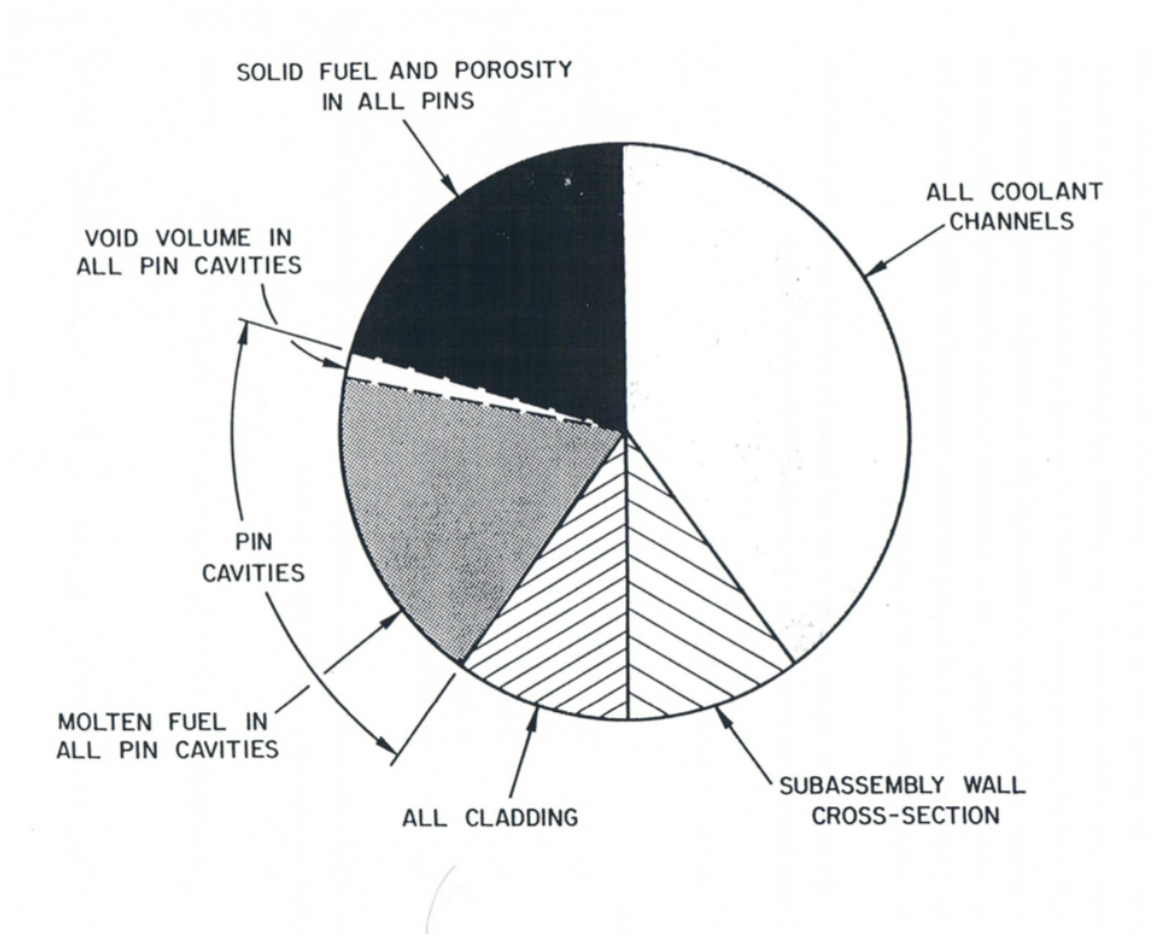


Figure 15.2.1: Illustration of the Generalized Volume Fraction

where the subscript **fsca** refers to the fission gas which is dissolved in the cavity fuel. The temperature T is the fuel temperature which should actually be written with subscript **fuca**. It should again be pointed out that the A's refer to total cross section areas of all the cavity fuel, free fission gas, and dissolved fission gas in the pins of one subassembly.

The generalized source or sink term is written as:

$$S' = \frac{S^{\updownarrow}}{\text{AXMX}} \tag{15.2-7}$$

where the source or sink term S_l represents a mass source or sink per unit time and unit length. The primed source or sink term has the dimension of mass per unit time and per unit smear volume. This unit of smear volume is a m^3 of the cell volume AXMX $\cdot \Delta z$ in which all relevant components (including components in all failed pins, the components in all the channels and the structure) are assumed to be uniformly smeared. From Eq. (15.2-7), a change in the generalized smear density, ρ', due to source or sink S' is just $S'\Delta t$.

15.2.3 Differential Equations for the In-Pin Fuel Motion and Description of Sink and Source Terms

The equation set for the in-pin fuel motion includes three mass conservation equations (for fuel, free fission gas, and dissolved fission gas), one energy, and one momentum conservation equation. The continuity equation for the molten fuel in the pin cavity is written:

$$\frac{\partial}{\partial \text{t}} \left(\rho_{\text{fuca}} A_{\text{fuca}}\right) = -\frac{\partial}{\partial \text{z}} \left(\rho_{\text{fuca}} A_{\text{fuca}} u_{\text{fuca}}\right) + S^{\updownarrow}_{\text{fuca,me}}\left(z,t\right) \tag{15.2-8}$$

where the subscript me refers to fuel melting into the pin cavities of all pins. By dividing by AXMX and using the definitions of the generalized smear densities and source and sink terms, Eq. (15.2-8) becomes:

$$\frac{\partial}{\partial \text{t}} \rho'_{\text{fuca}} = -\frac{\partial}{\partial \text{z}} \left(\rho'_{\text{fuca}} u_{\text{fuca}}\right) + S'_{\text{fuca,me}} \tag{15.2-9}$$

where the primed source is per unit time and per unit smear volume (see Eq. (15.2-7)). The integrated source term, $S'_{\text{fuca,me}}\Delta t_{\text{PN}}$, which is actually needed in the finite difference equations of the code, is calculated from

$$S'_{\text{fuca,me}}\Delta t_{\text{PN}} = \frac{\rho_{\text{fu,cabd}} \Delta A_{\text{me}} \text{NRPI}}{\text{AXMX}} \tag{15.2-10}$$

where

Δt_{PN} = PINACLE time step, s

$\rho_{\text{fu,cabd}}$ = Fuel density (including porosity) adjacent to the cavity boundary kg/m^3

ΔA_{me} = Area of fuel (including porosity) melted into the cavity per PINACLE time step per pin, m^3

NRPI = Number of pins per subassembly,

When the solid fuel node adjacent to the cavity has not yet exceeded an input melt fraction value FNMELT:

$$S'_{\text{fuca,me}}\Delta t_{\text{PN}} = 0 \tag{15.2-11}$$

The ΔA_{me} in Eq. (15.2-10) is related to a change in the cavity diameter by

$$\Delta A_{\text{me}} = \frac{\pi}{4} \Delta D_{\text{ca}} \left(2D_{\text{ca}} + \Delta D_{\text{ca}}\right) \tag{15.2-12}$$

where

D_{ca} = Cavity diameter, m

ΔD_{ca} = Change in the cavity diameter, m

In order to avoid adding the whole radial node instantaneously upon meeting the input melt fraction criterion FNMELT, the radial node is added gradually beginning at the time FNMELT is reached and the addition is

completed once the melt fraction has exceeded FNMELT + 0.1. Once the melt fraction has become greater than FNMELT, the change in cavity diameter is calculated from

$$\Delta D_{\text{ca}} = \text{FNMECA} \cdot 2 \cdot \Delta r_{\text{node}} \tag{15.2-13}$$

where Δr_{node} is the width of the heat-transfer node adjacent to the cavity wall before it is melted into the cavity. FNMECA is the fraction of this node width which has melted into the cavity per PINACLE time step.

$$\text{FNMECA} = \Delta t_{\text{PN}} \frac{\left[\frac{\left(T_{\text{cabd}}^{n+1} - T_{\text{cabd}}^{n}\right)}{\Delta t_{\text{Ht}}}\right]}{[0.1 \cdot (T_{\text{fu,liq}} - T_{\text{fu,sol}})]} = \frac{\Delta T_{\text{PN}}}{[0.1 \cdot (T_{\text{fu,liq}} - T_{\text{fu,sol}})]} \tag{15.2-14}$$

where

$T_{\text{cabd}}^{n}, T_{\text{cabd}}^{n+1}$ = The temperatures of the fuel node adjacent to the cavity at the beginning and at the end of the heat-transfer time step Δt_{Ht}, respectively

$T_{\text{fu,liq}}, T_{\text{fu,sol}}$ = Fuel liquidus and solidus temperatures, respectively (The difference between the two is the melting band width)

Δt_{PN} = The PINACLE time step.

ΔT_{PN} = The temperature change of the fuel adjacent to the cavity during a PINACLE time step

Eq. (15.2-14) implies that the whole heat-transfer node will be melted into the cavity after its temperature has risen by 1/10 of the melting band width beyond the input value **FNMELT**. Moreover, it is checked whether the neighboring solid fuel node also has exceeded the input melt fraction criterion FNMELT. If this is the case, the entire remaining node currently melting into the cavity is added immediately to the cavity. This situation can occur in TREAT experiments.

The free fission-gas mass conservation equation is written

$$\frac{\partial}{\partial \text{t}} \left(\rho_{\text{fica}} A_{\text{fica}}\right) = -\frac{\partial}{\partial \text{z}} \left(\rho_{\text{fica}} A_{\text{fica}} u_{\text{fuca}}\right) + S^{l}_{\text{fica,me}} + S^{l}_{\text{fsca,rl}} \tag{15.2-15}$$

where the subscripts **fsca** and **rl** refer to fission gas in solution and to the release of this dissolved fission gas, respectively. Dividing by AXMX and by using the definition of the generalized smear densities and source terms yields

$$\frac{\partial}{\partial \text{t}} \rho'_{\text{fica}} = -\frac{\partial}{\partial \text{z}} \left(\rho'_{\text{fica}} u_{\text{fuca}}\right) + S'_{\text{fica,me}} + S'_{\text{fsca,rl}} \tag{15.2-16}$$

The integrated source term for free fission gas due to fuel melt-in is similar to that in Eq. (15.2-10):

$$S'_{\text{fica,me}} \Delta t_{\text{PN}} = \rho_{\text{figb,cabd}} \Delta A_{\text{me}} \left(\frac{\text{NRPI}}{\text{AXMX}}\right) \tag{15.2-17}$$

where

ΔA_{me} = Calculated change in cavity cross sectional area from Eq. (15.2-12)

$\rho_{\text{figb,cabd}}$ = Density of the grain-boundary gas in the fuel node adjacent to the cavity

The quantity $\rho_{\text{figb,cabd}}$ is calculated from:

$$\rho_{\text{figb,cabd}} = \frac{\text{GNBFG2}_{\text{cabd}} \cdot \left(\frac{\text{FUMS}_{\text{cabd}}}{\text{FUELMS}_{\text{cabd}}}\right)}{\text{VOLUME}_{\text{cabd}}} \tag{15.2-18}$$

where

$GNBFG2_{cabd}$ = The grain-boundary fission-gas mass in the original radial fuel-pin node at the cavity boundary before it has actually melted in

$FUMS_{cabd}$ = The current fuel mass in the melting radial fuel-pin node at the cavity boundary

$FUELMS_{cabd}$ = The original fuel mass of the radial fuel-pin node at the cavity boundary before any fuel is removed due to melt-in

$VOLUME_{cabd}$ = The current volume of the melting radial fuel-pin node at the cavity boundary

The source term due to dissolved fission-gas release in Eq. (15.2-16) is:

$$S'_{\text{fsca,rl}} = \rho'_{\text{fsca}} \cdot \text{CIRTFS} \cdot \Delta t_{\text{PN}} \tag{15.2-19}$$

where

CIRTFS = Release time constant for dissolved fission gas which is input and has the dimensions s^{-1}

The same release constant is also used for the dissolved fission-gas release in the coolant channels - see Eq. (14.4-20). This is a relatively simple exponential decay-type approach to treat the release of the gas dissolved in molten fuel. However, the understanding of the mechanism of dissolved gas release from molten fuel is still very limited.

The dissolved fission-gas mass conservation equation is:

$$\frac{\partial}{\partial t}\left(\rho_{\text{fsca}} A_{\text{fuca}}\right) = -\frac{\partial}{\partial z}\left(\rho_{\text{fsca}} A_{\text{fuca}} u_{\text{fuca}}\right) + S'_{\text{fsca,me}} - S'_{\text{fsca,rl}} \tag{15.2-20}$$

By dividing this equation by AXMX and using the definitions of the generalized smear densities and sources, one obtains:

$$\frac{\partial}{\partial t}\left(\rho'_{\text{fsca}}\right) = -\frac{\partial}{\partial z}\left(\rho'_{\text{fsca}} u_{\text{fuca}}\right) + S'_{\text{fsca,me}} - S'_{\text{fsca,rl}} \tag{15.2-21}$$

where

$$S'_{\text{fsca,me}} = \rho_{\text{fi,ig,cadb}} \cdot \Delta A_{\text{me}} \cdot \left(\frac{\text{NRPI}}{\text{AXMX}}\right) \tag{15.2-22}$$

$\rho_{\text{fi,ig,cabd}}$ = The density of the intra-granular gas in the fuel node adjacent to the cavity.

The absolute value of the sink term due to the dissolved fission-gas release has been described in Eq. (15.2-19).

The fuel energy conservation equation is written:

$$\begin{aligned}\frac{\partial}{\partial t}\left[\rho_{\text{fuca}} e_{\text{fuca}} A_{\text{fuca}}\right] = &-\frac{\partial}{\partial z}\left[\rho_{\text{fuca}} e_{\text{fuca}} A_{\text{fuca}} u_{\text{fuca}}\right] + S^{l}_{\text{fuca,me}} e_{\text{fu,cabd}} \\ &+ Q A_{\text{fuca}} \rho_{\text{fuca}} - h_{\text{fuca,cabd}} \cdot \left(T_{\text{fuca}} - T_{\text{fu,cabd}}\right) \pi D_{\text{ca}} \text{NRPI}\end{aligned} \tag{15.2-23}$$

Dividing Eq. (15.2-23) by AXMX and using the definitions of the generalized smear densities and sources and sinks produces:

$$\begin{aligned}\frac{\partial}{\partial t}\left[\rho'_{\text{fuca}} e_{\text{fuca}}\right] = &-\frac{\partial}{\partial z}\rho'_{\text{fuca}} e_{\text{fuca}} u_{\text{fuca}} + S'_{\text{fuca,me}} e_{\text{fu,cabd}} \\ &+ Q\rho'_{\text{fuca}} - h_{\text{fuca,cabd}} \cdot \left(T_{\text{fuca}} - T_{\text{fu,cabd}}\right) \pi D_{\text{ca}} \frac{\text{NRPI}}{\text{AXMX}}\end{aligned} \tag{15.2-24}$$

By rewriting the left-hand side of Eq. (15.2-24) as

$$\frac{\partial}{\partial t}\left(\rho'_{\text{fuca}} e_{\text{fuca}}\right) = \rho'_{\text{fuca}} \frac{\partial}{\partial t} e_{\text{fuca}} + e_{\text{fuca}} \frac{\partial}{\partial t} \rho'_{\text{fuca}} \tag{15.2-25}$$

and by using the mass conservation Eq. (15.2-8), one obtains

$$\begin{aligned}\rho'_{\text{fuca}}\frac{\partial}{\partial \text{t}}e_{\text{fuca}} &= \frac{\partial}{\partial \text{t}}\left(\rho'_{\text{fuca}}e_{\text{fuca}}u_{\text{fuca}}\right) + e_{\text{fuca}}\frac{\partial}{\partial \text{z}}\left(\rho'_{\text{fuca}}u_{\text{fuca}}\right)\\ &+S'_{\text{fuca,me}}\left(e_{\text{fu,cabd}}-e_{\text{fuca}}\right)+Q\rho'_{\text{fuca}}\\ &-h_{\text{fuca,cabd}}\left(T_{\text{fuca}}-T_{\text{fu,cabd}}\right)\pi D_{\text{ca}}\frac{\text{NRPI}}{\text{AXMX}}\end{aligned} \tag{15.2-26}$$

where Q is the fission heat source per kg of fuel

$$\begin{aligned}Q = \text{FNPOHE}\cdot\text{POW}\cdot\text{PSHAPE}\left(K\right)\cdot\\ \left(1-\text{GAMSS}-\text{GAMTNC}-\text{GAMTNE}\right)\cdot\frac{F_{\text{POWER}}}{\text{FUMASS}\left(K\right)}\end{aligned} \tag{15.2-27}$$

In Eq. (15.2-27),

$$\text{FNPOHE} = \exp\left[\text{POWCOF}\left(1\right)+\Delta t_{\text{x}}\left\{\text{POWCOF}\left(2\right)+\Delta t_{\text{x}}\text{POWCOF}\left(3\right)\right\}\right] \tag{15.2-28}$$

where Δt_{x} is the time between the current *PINACLE* time and the beginning of the current main (point kinetics) time step. The POWCOF's are the coefficients that are found by fitting an exponential function to the power levels at the end of the last three point kinetics time step.

POW = steady-state power in the peak axial fuel-pin segment (see SAS4A input description)

PSHAPE (K) = ratio of pin power at axial node K to POW (see SAS4A input description)

GAMSS, GAMTNC, GAMTNE = fractions of total power for the direct heating of structure, coolant, and cladding, respectively. (See SAS4A input description)

FUMASS (K) = initial total fuel mass in axial pin segment K

F_{POWER} = power reduction factor if there is a radial power gradient in the pin (as is common in TREAT experiments):

$$F_{\text{POWER}} = \frac{\sum_{\text{I=1}}^{I_{\text{cabd}}}\text{PSHAPR}\left(I\right)\cdot\text{FUELMS}\ \left(I,K\right)\Big/\sum_{\text{I=1}}^{I_{\text{cabd}}}\text{FUELMS}\left(I,K\right)}{\sum_{\text{I=1}}^{\text{NT}}\text{PSHAPR}\ \left(I\right)\cdot\text{FUELMS}\left(I,K\right)\Big/\sum_{\text{I=1}}^{\text{NT}}\text{FUELMS}\left(I,K\right)} \tag{15.2-29}$$

where

NT = Number of radial pin nodes

I_{cabd} = Number of radial pin nodes in the cavity

PSHAPR (I) = Mass normalized radial power distribution in radial node I (See SAS4A input description)

FUELMS (I,K) = Initial fuel mass in the radial fuel-pin node I,K

The heat-transfer coefficient in Eq. (15.2-23) is the sum of a convective and a conduction heat-transfer term.

$$h_{\text{fuca,cabd}} = \left(h_1+h_2\right) \tag{15.2-30}$$

where

$$h_1 = \frac{k_{\text{fu}}}{D_{\text{ca}}}\cdot\text{St}\cdot\text{Pr}\cdot\text{Re} \tag{15.2-31}$$

This comes from the definition of the Nusselt number

$$\text{Nu} = \text{St}\cdot\text{Pr}\cdot\text{Re} \tag{15.2-32}$$

where

St = Stanton number = $\frac{h}{\rho u C_{\text{p}}}$

Pr = Prandtl number in $\frac{\mu C_{\text{p}}}{k}$

Re = Reynolds numbers = $\frac{D\rho u}{\mu}$

The Deissler correlation [15-7] was used for finding the relationship between the three nondimensional numbers on the right-hand side of Eq. (15.2-32). The Prandtl number for fuel is about 2.2. By using this value in the Deissler correlation, it can be shown that

$$\text{St} \approx 0.0158\,\text{Re}^{-0.2} \tag{15.2-33}$$

By using Eq. (15.2-33) and the definition of the Prandtl number,

$$h_1 = \frac{1}{D_{\text{ca}}} \cdot \mu_{\text{fu,liq}} \cdot C_{\text{P,fu}} \cdot \text{CIA3} \cdot \text{Re}_{\text{ca}}^{0.8} \tag{15.2-34}$$

where

k_{fu} = Conductivity of fuel which is input (CDFU)

$\mu_{\text{fu,liq}}$ = Liquid fuel viscosity which is input (VIFULQ)

CIA3 = Input constant. A value of 0.0158 is recommended because of Eq. (15.2-33)

$$Re_{\text{ca}} = \frac{\left(|u_{\text{fuca}}|\, D_{\text{ca}} \rho'_{\text{fuca}} \,/\theta_{\text{ca}}\right)}{\mu_{\text{fu,liq}}} \tag{15.2-35}$$

The conduction heat-transfer coefficient, h_2, which is relevant for a low flow or a stagnant flow condition is of the following form

$$h_2 = 4 \cdot \frac{k_{\text{fu}}}{D_{\text{ca}}} \tag{15.2-36}$$

The pressure calculation in the fuel-pin cavity is based on the assumption that the fission-gas pressure and fuel-vapor pressure can be added. The total cavity pressure is:

$$P_{\text{ca}} = P_{\text{fica}}\left(T_{\text{fuca}}, \rho_{\text{fica}}\right) + P_{\text{fvca}}\left(T_{\text{fuca}}\right) \tag{15.2-37}$$

If the input variable **INAPN** is equal to 1, then the sodium vapor pressure contribution will also be included in the pressure calculation, accounting for the possible presence of liquid sodium in metal fuel pins:

$$P_{\text{ca}} = P_{\text{fica}}\left(T_{\text{fuca}}, \rho_{\text{fica}}\right) + P_{\text{fvca}}\left(T_{\text{fuca}}\right) + P_{\text{Na}} \tag{15.2-38}$$

where the sodium vapor pressure contribution is calculated as follows:

$$P_{\text{Na}} = P_{\text{Nv,ca}}\left(T_{\text{fuca}}\right) - \left[P_{\text{fica}}\left(T_{\text{fuca}},\ \rho_{\text{fica}}\right) + P_{\text{fvca}}\right] \cdot \text{CINAPN} \tag{15.2-39}$$

where **CINAPN** is an input constant with values between 0 and 1.

The fission-gas pressure in Eq. (15.2-37) and Eq. (15.2-38) is calculated from a special form of the ideal-gas equation which takes the compressibility of the liquid fuel into account:

$$P_{\text{fica}} = \frac{R_{\text{fi}} \cdot T_{\text{fuca}} \rho'_{\text{fica}}}{\left(\theta_{\text{ca}} - \theta_{\text{fuca}} + \theta_{\text{fuca}} K_{\text{fu}} \cdot P_{\text{fica}}\right)} \tag{15.2-40}$$

where

R_{fi} = The universal gas constant divided by the molecular weight of the mixture of fission gas and helium fill gas (R_{fi} is equal to an input value RGAS which should take into account the relative amounts of krypton, xenon, and helium. The latter may be important for near-fresh fuel)

K_{fu} = Liquid fuel compressibility which is input (see CMFU)

The physically meaningful solution of the quadratic Eq. (15.2-40) is:

$$P_{\text{fica}} = \frac{-\left(\theta_{\text{ca}} - \theta_{\text{fuca}}\right) + \sqrt{\left(\theta_{\text{ca}} - \theta_{\text{fuca}}\right)^2 + 4\theta_{\text{fuca}} K_{\text{fu}} \cdot R_{\text{fi}} \cdot T_{\text{fuca}}\, \rho'_{\text{fica}}}}{2\theta_{\text{fuca}} K_{\text{fu}}} \tag{15.2-41}$$

There is also a second solution with a minus sign in front of the square root which does not give a physically meaningful result. Eq. (15.2-41) reduces to a single-phase liquid pressure solution for no fission gas and $\theta_{\text{fuca}} > \theta_{\text{ca}}$:

$$P_{\text{fica}} = -\frac{\left(\theta_{\text{ca}} - \theta_{\text{fuca}}\right)}{\left(\theta_{\text{fuca}} - K_{\text{fu}}\right)} \tag{15.2-42}$$

which is equivalent to the definition of the fuel compressibility. For void fractions greater than 30% the fission-gas pressure is calculated from a simplified form of Eq. (15.2-40) in which the term with K_{fu} is dropped.

The momentum conservation equation for the fuel/fission-gas mixture is:

$$\begin{aligned} \frac{\partial}{\partial \text{t}} \left(\rho_{\text{fuca}} A_{\text{fuca}} u_{\text{fuca}} + \rho_{\text{fica}} A_{\text{fica}} u_{\text{fuca}}\right) \\ = -\frac{\partial}{\partial \text{z}} \left(\rho_{\text{fuca}} A_{\text{fuca}} u_{\text{fuca}}^2 + \rho_{\text{fica}} A_{\text{fica}} u_{\text{fuca}}^2\right) \\ -A_{\text{ca}} \cdot \left(\frac{\partial \text{P}_{\text{ca}}}{\partial \text{z}}\right) - g\left(\rho_{\text{fuca}} A_{\text{fuca}} + \rho_{\text{fica}} A_{\text{fica}}\right) \\ -u_{\text{fuca}} \left|u_{\text{fuca}}\right| \cdot \left(\rho_{\text{fuca}} \frac{A_{\text{fuca}}}{A_{\text{ca}}} + \rho_{\text{fica}} \frac{A_{\text{fica}}}{A_{\text{ca}}}\right) A_{\text{ca}} \cdot F_{\text{friction}} / \left(2D_{\text{ca}}\right) \end{aligned} \tag{15.2-43}$$

Dividing Eq. (15.2-43) by AXMX and using the definitions for the generalized smear densities and mass sinks gives:

$$\begin{aligned} \frac{\partial}{\partial \text{t}} \left[u_{\text{fuca}} \left(\rho'_{\text{fuca}} + \rho'_{\text{fica}}\right)\right] \\ = \frac{\partial}{\partial \text{z}} \left[u_{\text{fuca}}^2 \left(\rho'_{\text{fuca}} + \rho'_{\text{fica}}\right)\right] \\ -\theta_{\text{ca}} \left(\frac{\partial \text{P}_{\text{ca}}}{\partial \text{z}}\right) - g\left(\rho'_{\text{fuca}} + \rho'_{\text{fica}}\right) \\ -u_{\text{fuca}} \left|u_{\text{fuca}}\right| \cdot \left(\rho'_{\text{fuca}} + \rho'_{\text{fica}}\right) \frac{F_{\text{friction}}}{\left(2D_{\text{ca}}\right)} \end{aligned} \tag{15.2-44}$$

where

u_{fuca} = Upward fuel velocity in the cavity (see the sign of the gravity head term in Eq. (15.2-43))

The Moody friction factor F_{friction} in Eq. (15.2-44) depends on the Reynolds number:

$$Re_{\text{pi}} = \frac{\left|u_{\text{fuca}}\right| D_{\text{ca}} \cdot \left(\rho'_{\text{fuca}} + \rho'_{\text{fica}}\right)}{\left(\theta_{\text{ca}} \cdot \text{VIFULQ}\right)} \tag{15.2-45}$$

where VIFULQ is the viscosity of liquid fuel which is input. The friction factor is

$$F_{\text{frication}} = \begin{cases} \frac{64}{\text{Re}_{\text{pi}}} & \text{for Re}_{\text{pi}} < \text{CIREFU} \\ \text{CIFRFU} & \text{for Re}_{\text{pi}} > \text{CIREFU} \end{cases} \tag{15.2-46}$$

where CIREFU and CIFRFU are both input and should be made consistent in order to avoid a jump in the friction factor at $\mathrm{Re}_{\mathrm{pi}} = \mathrm{CIREFU}$.

There is no term accounting for the fuel melt-in because this fuel is added to the cavity with zero axial velocity, and therefore does not change the total momentum. However, since the generalized fuel smear density will change in a cell with melt-in, this will lead to a velocity decrease in such a cell.

15.2.4 Finite Difference Equations for the In-Pin Motion Model

In the overview of the numerical scheme given in Section 15.2.1, it was pointed out that full donor cell spatial differencing and a largely explicit time differencing are used for treating the in-pin motion. The implicit aspect of the solution is that the mass and energy conservation equations are solved first and then a pressure is calculated on the basis of the mass and energy equation results. This advanced pressure is used in the momentum conservation equation.

The finite differencing of all the mass conservation equations is the same. The fuel mass conservation is used as an illustration.

$$\frac{\left(\rho'^{n+1}_{\text{fuca,K}} - \rho'^{n}_{\text{fuca,K}}\right)}{\Delta t_{\text{PN}}} = -\frac{\left((\rho' u)_{\text{fuca,K+1}} - (\rho' u)_{\text{fuca,K}}\right)}{\Delta z_{\text{K}}} + \sum_{\text{k}} S'_{\text{K,k}} \tag{15.2-47}$$

where

$$(\rho' u)_{\text{fuca,K}} = \begin{cases} \rho'_{\text{fuca,K-1}} u_{\text{K}} & \text{for } u_{\text{K}} > 0 \\ \rho'_{\text{fuca,K}} u_{\text{K}} & \text{for } u_{\text{K}} < 0 \end{cases} \tag{15.2-48}$$

u_{K} = Fuel velocity at the mesh cell boundary K.

$\rho'_{\text{fuca,K-1}}$ = Generalized fuel smear density at the mesh cell midpoint below boundary K

$\rho'_{\text{fuca,K}}$ = Generalized fuel smear density at the mesh cell midpoint above boundary K

The numerical grid used in the program was discussed in Section 15.2.1 which gives an overview of the numerical scheme. Densities are defined at the cell centers and velocities at the cell edges as illustrated in Figure 15.2.2. The source and sink terms have already been described in their finite difference form in the previous section.

The convective fluxes at the lower and upper boundaries of the cavity which are located in the end cells KK1 and KKMX, respectively, are

$$(\rho' u)_{\text{fuca,KK1}} = 0 \tag{15.2-49}$$

and

$$(\rho' u)_{\text{fuca,KKMX+1}} = 0 \tag{15.2-50}$$

The end cells are not always the same during a PINACLE run because the molten cavity can extend axially. When a new cell is added to the cavity, mass can only flow into this end cell if its cross section is at least 20% of that of the neighboring cell in the molten cavity. This restriction had to be included because of problems with overcompression of cells with a very small cross section.

The finite difference form of the fuel energy Eq. (15.2-26) is:

$$\begin{aligned} \frac{\rho'_{\text{fuca,K}}\left(e^{n+1}_{\text{fuca,K}} - e^{n}_{\text{fuca,K}}\right)}{\Delta t_{\text{PN}}} = &- \left[(\rho'_{\text{fuca}} e_{\text{fuca}} u_{\text{fuca}})^{n}_{\text{K+1}}\right. \\ &\left. - (\rho'_{\text{fuca}} e_{\text{fuca}} u_{\text{fuca}})^{n}_{\text{K}}\right] / \Delta z_{\text{k}} + e^{n}_{\text{fuca,K}} \left[(\rho'_{\text{fuca}} u_{\text{fuca}})^{n}_{\text{K+1}}\right. \\ &\left. - (\rho'_{\text{fuca}} u_{\text{fuca}})^{n}_{\text{K}}\right] / \Delta z_{\text{k}} + S'^{n}_{\text{fuca,me,K}} \left(e^{n}_{\text{fu,cabd,K}} - e^{n}_{\text{fuca,K}}\right) \\ &+ Q^{n}_{\text{K}} \rho'^{n}_{\text{fuca,K}} - h^{n}_{\text{fuca,cabd,K}} \left(T^{n}_{\text{fuca,K}} - T^{n}_{\text{fu,cabd,K}}\right) \pi D^{n}_{\text{ca,K}} \frac{\text{NRPI}}{\text{AXMX}} \end{aligned} \tag{15.2-51}$$

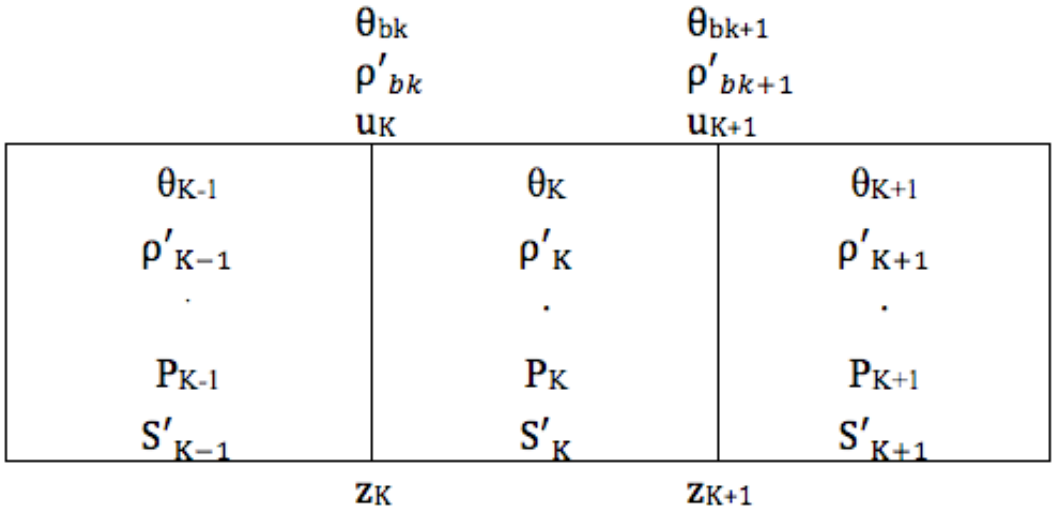


Figure 15.2.2: Numerical Grid Used in PINACLE

The fuel cavity temperature T_{fuca} in the above equation has to be calculated from the internal energy:

$$\text{For } e_{\text{fu,sol}} < e_{\text{fuca}} < e_{\text{fu,liq}}$$

$$T_{\text{fuca}} = T_{\text{fu,sol}} + (T_{\text{fu,liq}} - T_{\text{fu,sol}}) \cdot \frac{(e_{\text{fuca}} - e_{\text{fu,sol}})}{(e_{\text{fu,liq}} - e_{\text{fu,sol}})} \tag{15.2-52}$$

$$\text{For } e_{\text{fuca}} > e_{\text{fu,liq}}$$

$$T_{\text{fuca}} = T_{\text{fu,liq}} + \frac{(e_{\text{fuca}} - e_{\text{fu,liq}})}{C_{\text{p,fu}}} \tag{15.2-53}$$

where $C_{\text{p,fu}}$ is the fuel specific heat which is the single input value CPFU. The convective energy flux at cell boundary K in Eq. (15.2-51) is calculated as:

$$(\rho'_{\text{fuca}} e_{\text{fuca}} u_{\text{fuca}})_{\text{K}} = \begin{cases} (\rho'_{\text{fuca}} e_{\text{fuca}})_{\text{K-1}}\, u_{\text{fuca,K}} & \text{for } u_{\text{fuca,K}} > 0 \\ (\rho'_{\text{fuca}} e_{\text{fuca}})_{\text{K}}\, u_{\text{fuca,K}} & \text{for } u_{\text{fuca,K}} < 0 \end{cases} \tag{15.2-54}$$

At the lower and upper cavity ends, which are in cells KK1 and KKMX, the convective energy fluxes are zero:

$$(\rho'_{\text{fuca}} e_{\text{fuca}} u_{\text{fuca}})^n_{\text{KK1}} = 0 \tag{15.2-55}$$

and

$$(\rho'_{\text{fuca}} e_{\text{fuca}} u_{\text{fuca}})^n_{\text{KKMX+1}} = 0 \tag{15.2-56}$$

By using the convective fluxes from Eq. (15.2-48) and Eq. (15.2-54), and the definitions of the energy gain and loss terms given earlier, and by differencing the second term of Eq. (15.2-51) like the first, Eq. (15.2-51) can be solved for $e^{n+1}_{\text{fuca,K}}$. Fuel temperatures that are shown in the PINACLE output are calculated by using Eq. (15.2-52) and Eq. (15.2-53).

For the finite difference form of the momentum equation the following quantities have to be defined at the edges of the numerical cells: the combined fuel/fission-gas generalized smear density and the cavity volume fraction. These quantities become:

$$\rho'_{\text{fufi,bk}} = 0.5\left(\rho'_{\text{fuca,K-1}} + \rho'_{\text{fuca,K}}\right) + 0.5\left(\rho'_{\text{fica,K-1}} + \rho'_{\text{fica,K}}\right) \tag{15.2-57}$$

$$\theta_{\text{ca,bk}} = 0.5\left(\theta_{\text{ca,K}} + \theta_{\text{ca,K-1}}\right) \tag{15.2-58}$$

where the subscript **bk** indicates that these quantities are at the lower boundaries of cell K. This is shown in Figure 15.2.2. On the numerical grid the velocities are also defined on the cell boundaries, whereas the pressures, densities and volume fractions are defined at the cell centers. This is also shown in Figure 15.2.2. The finite difference form of the momentum conservation, Eq. (15.2-44), is written using Eq. (15.2-57) and Eq. (15.2-58) as:

$$\begin{aligned}\frac{\left(\rho'^{n+1}_{\text{fufi,bk}} u^{n+1}_{\text{fuca,K}} - \rho'^{n}_{\text{fufi,bk}} u^{n}_{\text{fuca,K}}\right)}{\Delta t_{\text{PN}}} = &-\left[\left(u^2_{\text{fuca}}\rho'_{\text{fufi}}\right)^n_{\text{K}}\right.\\ \left.-\left(u^2_{\text{fuca}}\rho'_{\text{fufi}}\right)^n_{\text{K-1}}\right]/\Delta z &- \theta_{\text{ca,bk}}\left[(1-\varepsilon)\cdot\left(P^n_{\text{ca,K}} - P^n_{\text{ca,K-1}}\right)\right.\\ &\left.+\varepsilon\cdot\left(P^{n+1}_{\text{ca,K}} - P^{n+1}_{\text{ca,K-1}}\right)\right]/\Delta z - g\rho'_{\text{fufi,bk}}\\ &- u^{n+1}_{\text{fuca,K}}\left|u^n_{\text{fuca,K}}\right|\rho'^{n}_{\text{fufi,bk}}\frac{F_{\text{friction,bk}}}{(2D_{\text{ca}})}\end{aligned} \tag{15.2-59}$$

where

ε = Input value EPCH that can be between zero and one (see Eq. (15.2-1))

$\Delta z = 0.5\left(\Delta z_{\text{K-1}} + \Delta z_{\text{K}}\right)$

By collecting all terms with u^{n+1}_{fuca} on the left-hand side of the equation, one obtains

$$\begin{aligned}u^{n+1}_{\text{fuca,K}}\left[\frac{\rho'^{n+1}_{\text{fufi,bk}}}{\Delta t_{\text{PN}}} + \left|u^n_{\text{fuca,K}}\right|\rho'^{n}_{\text{fufi,bk}}\frac{F_{\text{friction,bk}}}{(2D_{\text{ca}})}\right] &= \frac{\rho'^{n+1}_{\text{fufi,bk}} u^n_{\text{fuca,K}}}{\Delta t_{\text{PN}}}\\ &- \frac{\left[\left(u^2_{\text{fuca}}\rho'_{\text{fufi}}\right)^n_{\text{K}} - \left(u^2_{\text{fuca}}\rho'_{\text{fufi}}\right)^n_{\text{K-1}}\right]}{\Delta z}\\ &-\theta_{\text{ca,bk}}\left[(1-\varepsilon)\cdot\left(P^n_{\text{ca,K}} - P^n_{\text{ca,K-1}}\right)\right.\\ &\left.+\varepsilon\left(P^{n+1}_{\text{ca,K}} - P^{n+1}_{\text{ca,K-1}}\right) + \left(P^n_{\text{vi,K}} - P^n_{\text{vi,K-1}}\right)\right]\Delta z - g\rho'_{\text{fufi,bk}}\end{aligned} \tag{15.2-60}$$

The convective momentum flux in Eq. (15.2-60) is calculated as

$$\left(u^2_{\text{fuca}}\rho'_{\text{fufi}}\right)_{\text{K}} = \begin{cases}\rho'_{\text{fufi,K}} u^2_{\text{fuca,K}} & \text{if } \left(u_{\text{fuca,K}} + u_{\text{fuca,K+1}}\right) > 0\\ \rho'_{\text{fufi,K}} u^2_{\text{fuca,K+1}} & \text{if } \left(u_{\text{fuca,K}} + u_{\text{fuca,K+1}}\right) < 0\end{cases} \tag{15.2-61}$$

$$\left(u^2_{\text{fuca}}\rho'_{\text{fufi}}\right)_{\text{K-1}} = \begin{cases}\rho'_{\text{fufi,K-1}} u^2_{\text{fuca,K-1}} & \text{if } \left(u_{\text{fuca,K-1}} + u_{\text{fuca,K}}\right) > 0\\ \rho'_{\text{fufi,K-1}} u^2_{\text{fuca,K}} & \text{if } \left(u_{\text{fuca,K-1}} + u_{\text{fuca,K}}\right) < 0\end{cases} \tag{15.2-62}$$

where

$$\rho'_{\text{fufi,K}} = \rho'_{\text{fuca,K}} + \rho'_{\text{fica,K}} \tag{15.2-63}$$

The momentum fluxes for the lower and upper end cells of the cavity, which are designated by KK1 and KKMX, are:

$$\left(u^2_{\text{fuca}}\rho'_{\text{fufi}}\right)_{\text{KK1}} = \begin{cases}\rho'_{\text{fufi,KK1}} u^2_{\text{fuca,KK1+1}}\cdot 0.25 & \text{if } u_{\text{fuca,KK1+1}} > 0\\ \rho'_{\text{fufi,KK1}} u^2_{\text{fuca,KK1+1}} & \text{if } u_{\text{fuca,KK1+1}} < 0\end{cases} \tag{15.2-64}$$

$$\left(u^2_{\text{fuca}}\rho'_{\text{fufi}}\right)_{\text{KKMX}} = \begin{cases}\rho'_{\text{fufi,KKMX}} u^2_{\text{fuca,KKMX}} & \text{if } u_{\text{fuca,KKMX}} > 0\\ \rho'_{\text{fufi,KKMX}} u^2_{\text{fuca,KKMX}}\cdot 0.25 & \text{if } u_{\text{fuca,KKMX}} < 0\end{cases} \tag{15.2-65}$$

The factor 0.25 in the above convection terms comes from the assumption of a zero velocity at the end of the cavity.

The momentum Eq. (15.2-60) can be solved for $u_{\text{fuca,K}}^{n+1}$ if Eq. (15.2-57), Eq. (15.2-58), and Eq. (15.2-61) through Eq. (15.2-65) are used.

15.2.5 Treatment of the Fuel Ejection Above the Top of the Active Fuel Pin

The equations presented in the preceding sections apply to all the cavity cells before the cavity reaches the top fuel cell and the upward axial fuel ejection is initiated. When the fuel ejection begins, however, a different situation is created at the top of the fuel pin, through the creation of a new cell of variable length, as illustrated in Figure 15.2.3. The presence of the axial fuel ejection and of the variable length space above the fuel pin require a special treatment of the top cell, as outlined below.

15.2.5.1 Initiation of the Axial Fuel Ejection

When PINACLE is initiated it models a bottled-up cavity, extending axially from the cell KK1 to KKMX. Only limited fuel relocation occurs during this period, as the pressure gradients in the cavity are very small. For the initiation of the axial fuel motion two conditions must be satisfied: 1.) The cavity must extend to the upper active fuel cell, KCORE2, and 2.) the temperature of the top fuel pin surface must be higher than the fuel solidus temperature, i.e.:

$$T_{\text{fu,tp}} > T_{\text{fu,sol}} \tag{15.2-66}$$

The temperature of the top fuel surface $T_{\text{fu,tp}}$ is calculated as follows:

- if blanket pellets are present:

$$\begin{aligned} T_{\text{fu,tp}} &= T_{\text{fu}}\left(1, \text{KCORE2}+1\right) \\ &+ \left(T_{\text{fu}}\left(1, \text{KCORE2}\right) - T_{\text{fu}}\left(1, \text{KCORE2}+1\right)\right) \cdot \text{CIPNTP} \end{aligned} \tag{15.2-67}$$

- if no blanket pellets are present and the pin analyzed contains oxide fuel no axial temperature gradients are present at the fuel pin top and the constraint 15.2-55 is eliminated by using:

$$T_{\text{fu,tp}} = T_{\text{fu,liq}} \tag{15.2-68}$$

- if no blanket pellets are present and the pin analyzed contains metal fuel:

$$T_{\text{fu,tp}} = T_{\text{Na,plenum}} + \left(T_{\text{fu}}\left(1, \text{KCORE2}\right) - T_{\text{Na,plenum}}\right) \cdot \text{CIPNTP} \tag{15.2-69}$$

where:

CIPNTP is an input variable. The value 0.5, used in the TS-2 and M2 and M3 TREAT tests analyses has proven satisfactory in predicting the time of fuel motion onset

$T_{\text{fu}}\left(1, \text{KCORE2}\right)$ = The fuel temperature in the axial cell KCORE2 and radial cell 1 (i.e. the central cell)

After the initiation of the axial fuel ejection the PINACLE calculations are extended to the cell KKMXPN:

$$\text{KKMXPN} = \text{KCORE2} + 1 \tag{15.2-70}$$

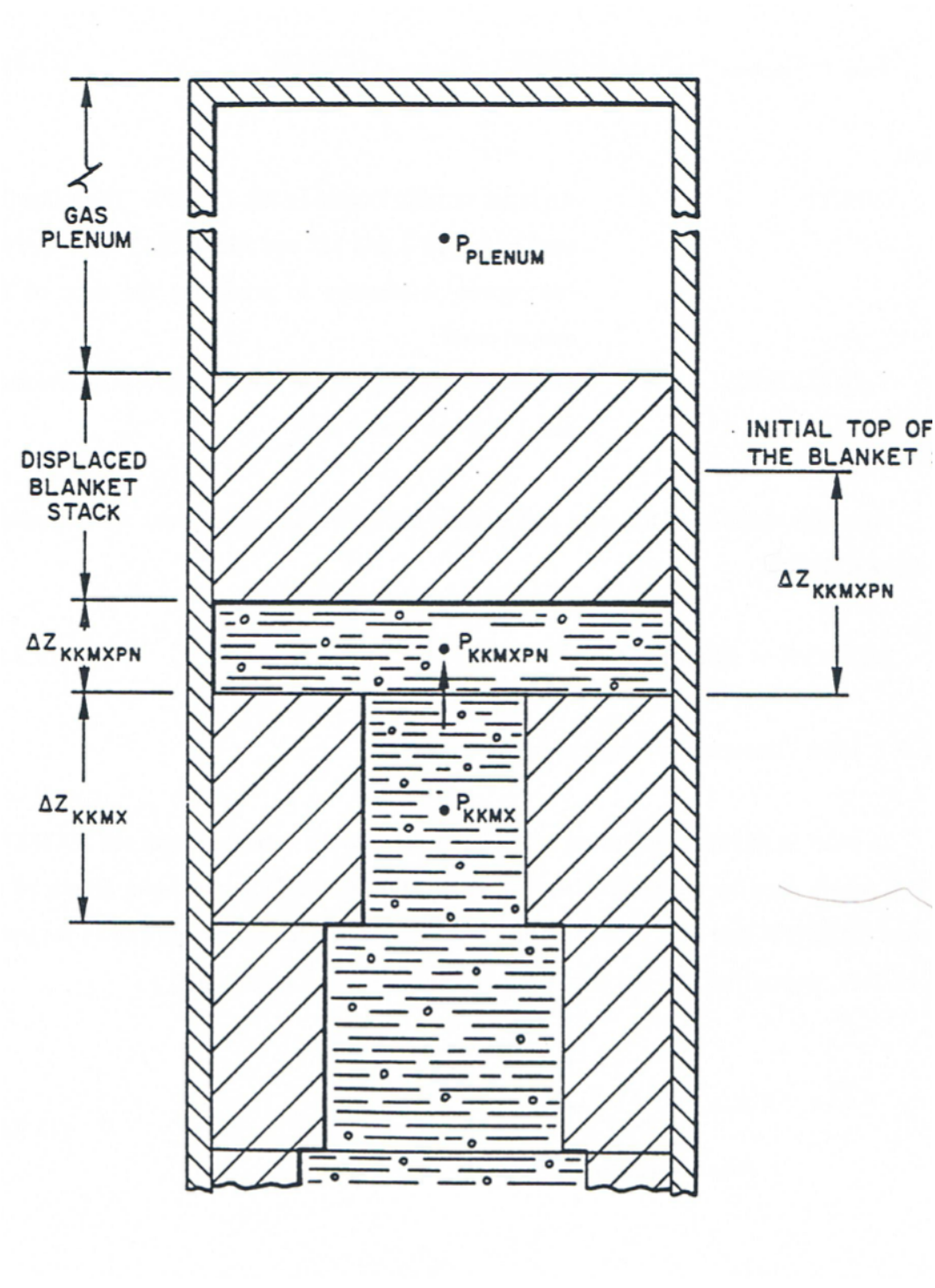


Figure 15.2.3: Geometry Used in the Calculation of the Fuel Ejection Above the Top of the Active Fuel

15.2.5.2 Mass Conservation Equation for the Top Cell

In order to minimize numerical complications due to the variable length cell KKMXPN, the cell length used in the mass conservation equation is $\Delta z^{\text{REF}}_{\text{KKMXPN}}$ the axial length of the segment KKMXPN (see Figure 15.2.3). The use of this fixed reference length allows the use of Eq. (15.2-47), unmodified, for the calculation of the $\rho^{n+1}_{\text{fu,ca,K}}$:

$$\rho'^{n+1}_{\text{fuca,K}} = \rho'_{\text{fuca,K}} - \left[(\rho' u)_{\text{fuca,K+1}} - (\rho' u)_{\text{fuca,K}}\right] \cdot \frac{\Delta t_{\text{PN}}}{\Delta z^{\text{REF}}_{\text{KKMXPN}}} + S_{\text{K}} \tag{15.2-71}$$

where:

$K = \text{KKMXPN} = \text{KCORE2} + 1$

S_{k} = The integrated source term $S'_{\text{K}} \cdot \Delta t_{\text{PN}}$, which is always zero in the KCORE2 $+ 1$ cell

$(\rho' u)_{\text{fuca,K+1}}$ = The convective mass flux at the boundary KKMXPN $+ 1$ is always zero

$$(\rho' u)_{\text{fuca,K}} = \begin{cases} \rho'_{\text{fuca,K-1}} \cdot u_{\text{K}} & \text{for } u_{\text{K}} > 0 \\ \rho'_{\text{fuca,K}} \cdot \frac{\Delta z^{\text{REF}}_{\text{KKMXPN}}}{\Delta z_{\text{KKMXPN}}} \cdot u_{\text{K}} \cdot \frac{\theta_{\text{K-1}}}{\theta_{\text{K}}} & \text{for } u_{\text{K}} < 0 \end{cases} \tag{15.2-72}$$

Note that the definition of the convective mass flux is more complex when $u_{\text{K}} < 0$. The smeared density $\rho'_{\text{fuca,K}}$ is multiplied by the factor:

$$\frac{\Delta z^{\text{REF}}_{\text{KKMXPN}}}{\Delta z_{\text{KKMXPN}}}$$

To obtain the actual smeared density in the existing cell, and the velocity u_{K}, which is defined at the boundary KKMXPN, but below the boundary, is multiplied by the factor.

$$\frac{\theta_{K-1}}{\theta_K}$$

To account for the abrupt area change present at the boundary KKMXPN $=$ KCORE2 $+ 1$.

15.2.5.3 The Energy Conservation Equation for the Top Cell

Due to the use of the reference cell $\Delta z^{\text{REF}}_{\text{KKMXPN}}$ conservation equation 15.2-43 remains virtually unchanged for the top cell KCORE2 $+ 1$:

$$\begin{aligned} e^{n+1}_{\text{fuca,K}} = e^{n}_{\text{fuca,K}} \left[(\rho'_{\text{fuca}} e_{\text{fuca}} u_{\text{fuca}})^n_{\text{K+1}} - (\rho'_{\text{fuca}} e_{\text{fuca}} u_{\text{fuca}})^n_{\text{K}}\right] \\ \cdot \frac{\Delta t_{\text{PN}}}{\Delta z^{\text{REF}}_{\text{KKMXPN}} \cdot \rho'_{\text{fuca,K}}} + e^{n}_{\text{fuca,K}} \cdot \left[(\rho'_{\text{fuca}} u_{\text{fuca}})^n_{\text{K+1}} - (\rho'_{\text{fuca}} u_{\text{fuca}})^n_{\text{K}}\right] \\ \cdot \frac{\Delta t_{\text{PN}}}{\Delta z^{\text{REF}}_{\text{KKMXPN}} \cdot \rho'_{\text{fuca,K}}} + \left[S'^{n}_{\text{fuca,me,K}} \cdot \left(e^{n}_{\text{fu,cabd,K}} - e^{n}_{\text{fuca,K}}\right)\right. \\ + Q^n_{\text{K}} \rho'^{n}_{\text{fuca,K}} - h^n_{\text{fuca,cabd,K}} \cdot \left(T^n_{\text{fuca,K}} - T^n_{\text{fu,cabd,K}}\right) \cdot \pi \cdot D^n_{\text{ca,K}} \\ \left. \cdot \frac{\text{NPRI}}{\text{AXMX}} \cdot \frac{\Delta z_{\text{KKMXPN}}}{\Delta z^{\text{REF}}_{\text{KKMXPN}}}\right] \cdot \frac{\Delta t_{\text{PN}}}{\rho'_{\text{fuca,K}}} \end{aligned} \tag{15.2-73}$$

where

$K = \text{KKMXPN} = \text{KCORE2} + 1$

$S'_{\text{fuca,me,K}}$ = The melting fuel source which is always zero in the top cell.

$(\rho'_{\text{fuca}} u_{\text{fuca}})_{\text{K}}$ = The convective mass term explained in Section 15.2.5.2.

$(\rho'_{\text{fuca}} e_{\text{fuca}} u_{\text{fuca}})_{\text{K+1}}$ = The convective energy term at boundary KKMXPN + 1 and is always zero

$(\rho'_{\text{fuca}} e_{\text{fuca}} u_{\text{fuca}})_{\text{K}}$ = The convective energy term at boundary KKMXPN and is defined below

$$(\rho' eu)_{\text{fuca,K}} = \begin{cases} (\rho'_{\text{fuca}} e_{\text{fuca}})_{\text{K-1}} \cdot u_{\text{fuca,K}} & \text{for } u_{\text{fuca,K}} > 0 \\ \rho'_{\text{fuca,K}} \cdot \frac{\Delta z^{\text{REF}}_{\text{KKMXPN}}}{\Delta z_{\text{KKMXPN}}} \cdot e_{\text{fuca,K}} \cdot u_{\text{fuca,k}} \cdot \frac{\theta_{\text{K-1}}}{\theta_{\text{K}}} & \text{for } u_{\text{fuca,k}} < 0 \end{cases} \tag{15.2-74}$$

The explanation of the correction terms $\frac{\Delta z^{\text{REF}}}{\Delta z_{\text{KKMXPN}}}$ and $\frac{\theta_{\text{K-1}}}{\theta_{\text{K}}}$ present in the convective energy term when $u_{\text{fuca,K}} < 0$ has been given in Section 15.2.5.2. Note also in Eq. (15.2-73) that the term describing the heat transfer between the molten fuel and the wall contains the correction factor:

$$\frac{\Delta z_{\text{KKMXPN}}}{\Delta z^{\text{REF}}_{\text{KKMXPN}}}$$

to account for the fact that the heat transfer surface is determined by the actual length of the top cell. Also for this top cell, the temperature $T_{\text{fu,cabd,K}}$ is set equal to the inner cladding temperature.

15.2.5.4 The Momentum Conservation Equation for the Top Cell

The momentum conservation equation in the top cell is used to calculate the fuel velocity $u_{\text{fuca,K}}$ at cell boundary $K = \text{KCORE2} + 1$. Because of the presence of the abrupt area change and of the blanket pellet stack and/or liquid sodium it was necessary to write a separate momentum equation for the top cell, rather than using the momentum equation developed to calculate the new fuel velocity at all other locations.

The geometry used to derive the momentum equation is illustrated in Figure 15.2.3. The momentum conservation cell is shown in Figure 15.2.3. The momentum conservation is written first in integral form.

$$\begin{aligned} \Delta \Bigg[u_{\text{fuca,K}} \cdot \rho_{\text{fufi}} \cdot A_{\text{ca,K-1}} \cdot \frac{\Delta z_{\text{K-1}}}{2} + u_{\text{fuca,K}} \cdot \frac{A_{\text{K-1}}}{A_{\text{K}}} \cdot (\rho_{\text{fufi}} A_{\text{ca,K}} \cdot \Delta z_{\text{KKMXPN}} \\ + M_{\text{slug}} \cdot \text{NRPI}) \Bigg] / \Delta t_{\text{PN}} = -A_{\text{bk}} \cdot (P_{\text{PLENUM}} - P_{\text{K-1}}) + \left(\rho \cdot u^2 \cdot A\right)_{\text{fuca,K}} \\ - \rho_{\text{fufi}} u_{\text{fuca,K}} \left| u_{\text{fuca,K}} \right| \cdot A_{\text{K-1}} F_{\text{FRICTION,K-1}} \cdot \frac{\Delta z_{\text{K-1}}}{4 D_{\text{ca,K-1}}} \\ - \rho_{\text{fufi}} u_{\text{fuca,K}} \left| u_{\text{fuca,K}} \right| \cdot A_{\text{K}} \left(\frac{A_{\text{K-1}}}{A_{\text{K}}} \right)^2 \cdot F_{\text{FRICTION,K}} \cdot \frac{\Delta z_{\text{KKMXPN}}}{2 D_{\text{ca,K}}} \end{aligned} \tag{15.2-75}$$

Note that in writing Eq. (15.2-75) the blanket stack and/or liquid sodium was assumed to move with the same velocity as the fuel in the top cell KCORE2 + 1. More will be said about this assumption later in this chapter. The mass of the fuel stack and or sodium slug is defined by the input variable **FUSLMA**. After dividing Eq. (15.2-75) by AXMX, multiplying by the number of pins NRPI, and using the definition of generalized smear densities and volume fractions we obtain:

$$\begin{aligned} \frac{\Delta z_{\text{K-1}}}{2} \cdot \Delta \left[u_{\text{fufi,K}} \cdot \rho'_{\text{fufi,K-1}} \right] / \Delta t_{\text{PN}} + \frac{\theta_{\text{ca,K-1}}}{\theta_{\text{ca,K}}} \cdot \Delta \Bigg[u_{\text{fuca,K}} \Big(\rho'_{\text{fufi,K}} \cdot \Delta z^{\text{REF}}_{\text{KKMXPN}} \\ + \frac{M_{\text{slug}} \cdot \text{NRPI}}{\text{AXMX}} \Big) \Bigg] / \Delta t_{\text{PN}} = -\theta_{\text{bk}} \cdot (P_{\text{PLENUM}} - P_{\text{K-1}}) + \left(\rho' u^2\right)_{\text{fufi,K}} \\ - \rho'_{\text{fufi,K-1}} \cdot u_{\text{fuca,K}} \cdot \left| u_{\text{fuca,K}} \right| \cdot F_{\text{FRICTION,K-1}} \cdot \frac{\Delta z_{\text{K-1}}}{4 \cdot D_{\text{ca,K-1}}} \\ - \rho'_{\text{fufi,K}} \cdot u_{\text{fuca,K}} \cdot \left| u_{\text{fuca,K}} \right| \cdot \left(\frac{\theta_{K-1}}{\theta_K} \right)^2 \cdot F_{\text{FRICTION,K}} \cdot \frac{\Delta z_{\text{KKMXPN}}}{2 D_{\text{ca,K}}} \end{aligned} \tag{15.2-76}$$

The convective momentum flux $\left(\rho' u^2\right)_{\text{fufi,K}}$ is calculated using Eq. (15.2-61) and the area fraction θ_{bk} at the abrupt area change is calculated as follows:

$$\theta_{\text{bk}} = \begin{cases} \theta_{\text{K}} & \text{if } u_{\text{K}} \geq 0 \text{ i.e. expansion} \\ 1.67 \frac{\theta_{\text{K}} \cdot \theta_{\text{K-1}}}{\theta_{\text{K}} + \theta_{\text{K-1}}} & \text{if } u_{\text{K}} < 0 \text{ i.e. contraction} \end{cases} \tag{15.2-77}$$

The friction factor F_{FRICTION} is defined by Eq. (15.2-46) and 15.2-39a. By using the identity:

$$\Delta\left(u\rho'\right) = \rho'^{n+1} \cdot \Delta u + u^n \cdot \Delta\rho' \tag{15.2-78}$$

Eq. (15.2-76) is now rearranged in the final form:

$$\begin{aligned} &\frac{\Delta u_{\text{fufi,K}}}{\Delta t_{\text{PN}}} \cdot \rho'_{\text{fufi,K-1}} \cdot \frac{\Delta z_{\text{K-1}}}{2} + \rho'_{\text{fufi,K}} \cdot \Delta z^{\text{REF}}_{\text{KKMXPN}} \cdot \frac{\theta_{\text{ca,K-1}}}{\theta_{\text{ca,K}}} + \frac{\theta_{\text{ca,K-1}}}{\theta_{\text{ca,K}}} \\ &\cdot \frac{M_{\text{slug}} \cdot \text{NRPI}}{\text{AXMX}} \cdot C_{\text{TOP}} = -\theta_{\text{bk}} \cdot P_{\text{PLENUM}} \cdot C_{\text{TOP}} + P_{\text{K}}\left(1 - C_{\text{TOP}}\right) - P_{\text{K-1}} \\ &\qquad + \left(\rho' u^2\right)_{\text{fufi,K}} - \rho'_{\text{fufi,K-1}} \cdot u_{\text{fuca,K}} \cdot \left|u_{\text{fuca,K}}\right| \cdot F_{\text{FRICTION,K-1}} \\ &\cdot \frac{\Delta z_{\text{K-1}}}{4 \cdot D_{\text{ca,K-1}}} - \rho'_{\text{fufi,K}} \cdot u_{\text{fuca,K}} \cdot \left|u_{\text{fuca,K}}\right| \cdot \left(\frac{\theta_{\text{K-1}}}{\theta_{\text{K}}}\right)^2 \cdot F_{\text{FRICTION,K}} \cdot \frac{\Delta z_{\text{KKMXPN}}}{2 D_{\text{ca,K}}} \end{aligned} \tag{15.2-79}$$

Eq. (15.2-79) is used to calculate the change in velocity $\Delta u_{\text{fufi,K}}$ at the top cavity boundary and thus determines the new fuel ejection velocity. Note that Eq. (15.2-79) contains the variable CTOP, which was not present in Eq. (15.2-77). This variable has initially the value 1, so the Eq. (15.2-79) is identical to Eq. (15.2-78). However, when the solid pellet stack reaches a rigid obstacle, i.e., when $\Delta z_{\text{KKMXPN}} = \texttt{FUSLDT}$, the variable C_{TOP} is set to zero.

This has the effect of decoupling the slug from molten fuel motion. The slug is assumed to remain fixed in place. It cannot move upwards because of the rigid obstacle, such as dimples, and presumably it cannot move downwards because of fuel freezing and crust formation in the space underneath. When $C_{\text{TOP}} = 0$ the pressure difference used in the momentum equation thus become $P_{\text{K}} - P_{\text{K-1}}$ rather than the $P_{\text{PLENUM}} - P_{\text{K-1}}$ used previously.

15.2.6 Time-step Determination for the Pre-Failure In-Pin Motion

The PINACLE time step Δt_{PN} used in the numerical solution of all the in-pin and channel conservation equations is restricted by the sonic Courant condition for the in-pin flow. The determination of the PINACLE time step, Δt_{PN}, is given at the end of the channel flow description in Section 15.3.1.3. In the present section, only the restriction imposed by the sonic Courant condition for the in-pin flow is described.

The time-step for the in-pin motion is computed to be a fraction, 0.4, of the minimum time step based on the sonic Courant condition

$$\Delta t_{\text{PN}} = 0.4 \cdot \min\left[\Delta z_{\text{K}} / \left(v_{\text{sonic,K}} + \left|u_{\text{fuca,K}}\right|\right)\right]_{\text{K=KK1,KKMX}} \tag{15.2-80}$$

The minimum in Eq. (15.2-80) is evaluated over all the axial cells of the molten fuel cavity. The sonic velocity, v_{sonic}, is calculated from an expression for an adiabatic, homogeneous two-component gas-liquid mixture which is based on Eq. 27 in Reference [15-8].

$$\begin{aligned} v^2_{\text{sonic}} = &\frac{\gamma_{\text{fi}} P_{\text{fica}}}{\left[\alpha^2_{\text{fica}} \cdot \rho_{\text{fica}} + \alpha_{\text{vica}}\left(1 - \alpha_{\text{fica}}\right)\rho_{\text{fuca}}\right]} \\ &+ \left[\left(1 - \alpha_{\text{fica}}\right)^2 \rho_{\text{fuca}} + \alpha_{\text{fica}}\left(1 - \alpha_{\text{vica}}\right)\rho_{\text{fica}}\right] \gamma_{\text{fi}} P_{\text{fica}} K_{\text{fu}} \end{aligned} \tag{15.2-81}$$

where

$\alpha_{\text{fica}} = \frac{\theta_{\text{fica}}}{\theta_{\text{ca}}}$ = Void fraction in the cavity

$\gamma_{\text{fi}} = \frac{C_{\text{p,fi}}}{C_{\text{v,fi}}} = 1.4$ (value assumed in PINACLE)

K_{fu} = CMFU = Input liquid fuel compressibility

The above equation holds for adiabatic gas behavior, although the in-pin fission-gas treatment in PINACLE is isothermal (the gas temperature is assumed to be always equal to the fuel temperature). However, the sonic velocity for adiabatic gas behavior is higher than that for isothermal gas behavior, and thus, leads to a more conservative (i.e., smaller) time step. Moreover, if a pure gas flow were treated in sections of pins with a prefabricated central hole, the current time-step determination would actually be necessary.

15.3 Computer Implementation

15.3.1 Detailed Logic Flow Description

This section describes in detail the logical sequence of the solution method used in PINACLE. The structure of the PINACLE driver as well as initiation and interfacing considerations have been presented in Section 15.1.2. Some information about the interaction between LEVITATE models was presented in Section 15.1.2.1 and the method of solution for the molten cavity model was described in Section 15.2.1. This section will use, and occasionally repeat some of this information in order to present a comprehensive picture of the general solution method. Some considerations about data management and time-step selection are also presented.

15.3.1.1 Data Management Considerations

Every time PINACLE begins calculations in a channel, the permanently stored information is retrieved from the data container and loaded in the common blocks. This operation is performed in the TSTHRM module. A number of arrays and other data, however, which can be calculated from the permanently stored variables, are not retained in the permanent storage. These data are recalculated in the routine PNSET2 every time PINACLE receives control in a certain channel, i.e., at the beginning of a heat transfer time step. These variables are stored in temporary common blocks, and kept only as long as PINACLE retains control in the channel. At the end of the heat transfer time step, when PINACLE returns control to TSTHRM, these variables are lost. Only the permanent common blocks are saved in the data container. This procedure is used in order to reduce the amount of storage required for SAS4A. The permanent PINACLE variables are stored in the block PLUC, which is also used by PLUTO2 and LEVITATE. Thus, when PLUTO2/LEVITATE are initiated, the cavity data calculated by PINACLE are automatically available for these models.

15.3.1.2 Logic Flow for Solution Advancement

As explained in Section 15.1.2.1 the PINACLE model describes the thermal-hydraulic events that occur inside the fuel pin cavity. During each time step PINACLE advances the solution by calculating the value of all variables at the end of the time step. The time-step selection is described in Section 15.2.6. The PINACLE calculation begins by advancing the solution for the radial cavity extension in each axial cell. This calculation is performed in the PN1PIN routine. This routine calculates the new fuel and fission gas masses in each axial cell and solves the mass conservation equations. Then the energy conservation equation is solved in each axial cell, providing the new fuel and fission gas temperature. Using the updated masses, temperatures and volumes, the PN1PIN subroutine calculates the new local pressures. The velocities are then calculated in the PN2PIN routine, which solves the momentum conservation equations for each staggered momentum cell. A separate momentum equation is used to describe the ejection of molten fuel into the upper plenum if this ejection has been initiated. PN2PIN also prints the regular PINACLE output if necessary.

Using the fuel distribution at the end of each time step PINACLE then calculates the new channel fuel reactivity. At the end of each primary step the axial fuel mass distribution is returned to the FEEDBK routine, as explained in Section 15.1.3.3.3. Finally, when the end of the heat-transfer time step is reached, PINACLE calls the routine HTRVPN, which calculates the new temperatures in the solid fuel pin at all axial locations. These temperatures will be used in the next time step in the PN1PIN routine to calculate the new cavity diameter and the heat flux between the molten fuel and the cavity wall.

15.3.1.3 Time-step Considerations

The PINACLE driver routine, PINACL is called by the SAS4A transient driver TSTHRM at the beginning of each heat-transfer time step during a primary loop time step. The primary time step is common for all SAS4A calculation channels and is always smaller than or equal to the main time step used in the point kinetics calculation.

When in-pin fuel motion is detected in a given channel, the PINACLE time, TIMEPN, is set to zero in the CAVMOT routine. TIMEPN is initially advanced by adding the PINACLE minimum time step DTPNIN to the time TIMEPN. Subsequently, the time TIMEPN will be advanced by the time step calculated by PINACLE, as described below.

A hydrodynamic time step DTPI, is calculated first for the in-pin hydrodynamic model, as outlined in Section 15.2.6. This time step is then compared with the present maximum value 1.10^{-3} s which is used to avoid numerical heat transfer instabilities. The smallest value is retained as the PINACLE time step. This value is further compared with the input minimum time step, DTPNIN and the larger value is retained. Finally, the PINACLE time step can be cut back if the newly determined time:

$$t_{\mathrm{PN}}^{n+1} = t_{\mathrm{PN}}^{n} + \Delta t$$

exceeds the end of the heat-transfer time step. In this case, the new PINACLE time step will be reduced so that the new PINACLE time t_{PN}^{n+1} will coincide with the end of the heat-transfer time step.

As shown in the flow diagram 15.1-5, PINACLE retains control and advances the solution in a given channel until the end of a heat transfer time step is reached.

15.3.1.4 List of PINACLE Routines

This section contains a list of all the subroutines that are part of the PINACLE model:

Name	Description
CAVMOT	Decides when to initiate PINACLE calculations. This routine is called at the end of each heat transfer time step from DFORM3.
PNINIT	Sets several important PINACLE variables and calls the initialization routines PNIPT and PNSET. This routine is called from CAVMOT.
PNINPT	This routine is used together with PNSET to initialize the PINACLE variables. It is called from PNINIT.
PNSET	This routine is used, together with PNINPT, to initialize the PINACLE variables. It is called from PNINIT.
PINACL	PINACLE driver routine. It is called at the end of each heat transfer time step from TSTHRM.
PN1PIN	Solves the mass and energy conservation equations in the pin cavity. Also calculates the radial cavity expansion. It is called each PINACLE time step from PINACL.
PN2PIN	Solves the momentum conservation equations in the pin cavity. Also determines the next PINACLE hydrodynamic time-step and prints the regular PINACLE output. It is called every PINACLE time step from PINACL.
HTRVPN	Heat transfer routine active after PINACLE initiation. Calculates the transient temperatures in the solid fuel pin and cladding. It is called from PINACL at the end of each heat transfer time step.

15.3.2 Input Parameters Relevant to PINACLE

The input parameters relevant to PINACLE are summarized in Table 15.3.1. The description of these parameters has also been included in the SAS4A input listing. Table 15.3.1 lists the recommended values for these parameters and the sections and equations where those parameters are mentioned in the text. The list of equations is not necessarily exhaustive, and some input parameters might appear in other equations, in addition to those listed in the table (e.g., AXMX appears in many places and it was not possible to list all occurrences).

Table 15.3.1: PINACLE Input Description

Input Loc.	Var.	Sect.	Eq.	Sug. Value (MKS)	Comments
Block 1, INPCOM					
49	INAPN	15.2.3		0	If INAPN = 1 then PINACL will assume that small amounts of liquid sodium are present to each axial cell in the cavity. The sodium vapor pressure will be added to the local pressure, depending on the value of CINAPN (13/1286). This input is now provided for parametric studies. The modeling of the sodium field in the cavity will be added to PINACLE at a later time. If INAPN = 0 no sodium pressure is added to the cavity pressure.

continues on next page

Table 15.3.1 – continued from previous page

Input Loc.	Var.	Sect.	Eq.	Sug. Value (MKS)	Comments
Block 13, PMATCM					
1277	DTPNIN	15.3.1.3		2×10^{-5}	Minimum time step.
1279	DTPNP	15.3.3.1		0.05	Determines the time interval between PINACLE printouts.
1286	CINAPN	15.2.3	Eq. (15.2-39)	1	Determines the fraction of the sodium vapor pressure to be added to the local pressure. Can have values between 0 and 1. If CINAPN = 0 all the local sodium vapor pressure is added to the local pressure. If CINAPN = 1 then only the excess P_{na} (T:sub:*na*) - P_{local} is added to the local pressure.
1295	CIPNTP	15.2.5.1	Eq. (15.2-69)	0.5	This variable controls the calculation of the fuel pin top boundary temperature, which is important in triggering the axial in-pin fuel relocation. Can vary between 0 and 1. If 1, the boundary temperature is the same as the temperature of the central top fuel mode. If 0, the boundary temperature is equal to the temperature of the material in the mode above the active fuel, i.e. sodium or blanket fuel central mode.
1296	ROGSPI			17.74 for U-Fissium	Mass of fission gas generated in the fuel pin per unit volume of the original pin and percent burnup. Kg/m^3 fuel/% B.U.
Block 51, INPCHN					
190	IPNPLT	15.3.3.2		1	Controls the fuel distribution printer plot from PINACLE. Can be 0 or 1. If 1, the printer plot will be printed with each PINACLE printout.
194	IPNGO	15.3.3.1			The variables IPNGO, IPNSTP and IPNNEW can be used to obtain more frequent PINACLE output between the cycles IPNGO and IPNSTP. The PINACLE output will be printed every IPNNEW cycles.
195	IPNSTP	15.3.3.1			
196	IPNNEW	15.3.3.1			
Block 65, FUELIN					

continues on next page

Table 15.3.1 – continued from previous page

Input Loc.	Var.	Sect.	Eq.	Sug. Value (MKS)	Comments
22	FPINAC	15.1.2.1		0.2	Controls the PINACLE initiation. The PINACLE calculations can only start in a channel where the maximum areal melt fraction (defined as FNMELT times the heat of fusion) is greater than FPINAC. See also CPINAC (65/23) and FNMELT (13/1169).
23	CPINAC	15.1.2.1		0.5	In addition to the FPINAC condition, PINACLE will start only if there are at least 3 adjacent cells with a melt fraction larger than CPINAC * FPINAC.
52	FUSLDT	15.2.5.4		Geometry dependent	Specified the length of possible free motion for the blanket fuel stack and/a liquid sodium. If only liquid sodium is present it should be set equal to the plenum length.
53	FUSLMA	15.2.5.4	Eq. (15.2-75)		Specifies the mass of the blanket pellet stack and/or sodium which will have to be displaced when the axial fuel relocation begins. This mass refers to *one* pin only.
54	BURNFU				Specifies the atom percent burnup for the fuel pins in the channel. It is used in the fission gas calculations when DEFORM-4 is not active.

15.3.3 Output Description

15.3.3.1 Regular Output

The PINACLE output has been designed to provide the essential information about the fuel pins in the channel at a given point in time. The output is printed from PINACL and can be obtained at equal time intervals, by specifying the input integers IPNGO, IPNSTP and IPNNEW. The PINACLE output will then be printed between cycles IPNGO and IPNSTP every IPNNEW cycles.

The regular PINACLE output is illustrated in Figure 15.3.1. The first line in the pin-related output contains the computational cycle number ICYCLE, the SAS4A channel number ICH and the current time, TIMEPN. This time is measured from the initiation of the in-pin fuel motion calculations in the given channel.

The second line contains some summary information about the fuel pin cavity and the status of the axial fuel ejection above the active fuel column. All the masses in this line refer to the whole subassembly:

Variable	Description
FUSLDB	Length of molten fuel column ejected above the active fuel pin, m.
SMFUCA	Total mass of molten fuel in the pin cavity, kg.
SMFICA	Total mass of free fission gas in the pin cavity, kg.
SMFSCA	Total mass of dissolved fission gas in the pin cavity, kg.
SMFUME	Total mass of pin fuel that has molten since PINACLE initiation, kg.
SMFIME	Total mass of free fission gas released to the cavity due to fuel melting, kg.
SMFSRT	Total mass of fission gas that was originally dissolved in the molten fuel but was released in the meantime, kg.
TEPNTP	Temperature of the active fuel column top interface. The axial fuel ejection of the fuel from the cavity into the plenum is initiated only after this temperature reaches the fuel solidus temperature, K.
PRPLNM	Current pressure in the pin plenum (Pa). If the blanket stack did not reach a rigid obstacle, i.e. FUSLDB < FUSLDT, the difference between this pressure and the pressure in the top cavity cell controls the axial fuel ejection.

Two groups of columns follow, providing more detailed information for all axial cells in the cavity. These columns are described below. Whenever masses are involved they refer to a single pin, rather than to a whole subassembly.

First group of columns:

Variable	Description
K	The number of the axial cell in the cavity. This number refers to the cavity grid, which differs from the channel grid by the integer IDIFF, i.e., K=I - IDIFF.
DICA	Diameter of the cavity, m.
FUELSD	Molten fuel density smeared over the cavity area, kg/m^3.
FUELM	Molten fuel mass, kg.
RHFUCA	Physical density of the molten fuel, kg/m^3.
FISGSD	Free fission-gas density, smeared over the cavity area, kg/m^3.
FISGM	Free fission-gas mass, kg.
FISGDM	Dissolved fission-gas mass, kg.
FNFIGB	Fraction of the fission gas which is released instantaneously upon fuel melting. This fraction is currently an input constant, independent of the axial location.
EGFUCA	Enthalpy of the molten fuel in the cavity, J/kg.
TEFUPI	Temperature of the molten fuel, K.

The second group of columns:

ICYCLE= 3136 ICH= 2 TIMEFN= 0.17502711D+01 DTFNC= 0.67000000D-03

FUEL PIN INFORMATION

FUSLDB	SMFUCA	SMFICA	SMFSCA	SMFUME	SMFIME	SMFSRT	TEFNTP	PRPLNN
0.062	0.29575467D+02	0.52700572D-01	0.20571169D-03	0.11170189D+02	0.60643511D-02	0.36405116D-01	0.13792925D+04	0.36089787D+07

K	DICA	FUELSD	FUELM	RHFUCA	FISGSD	FISGM	FISGDM	FNFIGB	EGFUCA	TEFUPI
8	3.099D-04	5.339D+03	1.818D-05	1.475D+04	1.870D+01	6.370D-03	3.293D-18	1.500D-01	1.225D+05	9.726D+02
9	1.917D-03	1.011D+04	1.335D-03	1.475D+04	1.865D+01	2.463D-06	5.193D-15	1.500D-01	1.243D+05	9.801D+02
10	2.755D-03	1.016D+04	2.771D-03	1.475D+04	1.827D+01	4.981D-06	1.061D-14	1.500D-01	1.262D+05	9.877D+02
11	3.369D-03	1.027D+04	4.184D-03	1.475D+04	1.765D+01	7.187D-06	1.561D-14	1.500D-01	1.278D+05	9.944D+02
12	3.939D-03	1.040D+04	5.916D-03	1.475D+04	1.683D+01	9.573D-06	2.065D-14	1.500D-01	1.301D+05	1.004D+03
13	3.986D-03	1.043D+04	5.961D-03	1.474D+04	1.613D+01	9.170D-06	2.457D-14	1.500D-01	1.353D+05	1.024D+03
14	3.980D-03	1.053D+04	5.990D-03	1.474D+04	1.564D+01	8.895D-06	7.116D-13	1.500D-01	1.399D+05	1.043D+03
15	4.592D-03	1.055D+04	7.984D-03	1.474D+04	1.554D+01	1.177D-05	8.900D-11	1.500D-01	1.415D+05	1.049D+03
16	4.592D-03	1.037D+04	7.859D-03	1.474D+04	1.600D+01	1.211D-05	7.960D-09	1.500D-01	1.449D+05	1.063D+03
17	5.203D-03	9.751D+03	9.481D-03	1.474D+04	1.785D+01	1.735D-05	6.526D-07	1.500D-01	1.497D+05	1.082D+03
18	5.203D-03	9.766D+03	9.496D-03	1.474D+04	1.803D+01	1.753D-05	1.495D-08	1.500D-01	1.464D+05	1.068D+03
19	5.083D-03	9.731D+03	9.026D-03	1.474D+04	1.774D+01	1.645D-05	5.295D-08	1.500D-01	1.510D+05	1.087D+03
20	5.033D-03	9.646D+03	8.799D-03	1.473D+04	1.792D+01	1.633D-05	9.427D-09	1.500D-01	1.537D+05	1.098D+03
21	5.203D-03	9.184D+03	8.929D-03	1.473D+04	1.943D+01	1.839D-05	1.965D-08	1.500D-01	1.546D+05	1.102D+03
22	3.414D-03	7.767D+03	3.251D-03	1.473D+04	2.405D+01	1.007D-05	1.316D-09	1.500D-01	1.588D+05	1.118D+03
23	6.121D-03	1.002D+04	1.815D-02	1.474D+04	1.747D+01	3.164D-05	1.238D-10	1.500D-01	1.458D+05	1.066D+03

K	ZZPI	UFPI	PRCA	PRFIPI	PRFVPI	FUVOFR	FUMESH	FIMESH	FALRAT	PRHACA
8	3.302D-01	1.000D-12	1.814D+06	1.814D+06	7.799D-19	3.620D-01	0.0	3.479D-01	7.078D-08	0.0
9	3.759D-01	1.000D-12	3.696D+06	3.696D+06	1.410D-18	6.854D-01	0.0	3.145D-01	1.585D-07	0.0
10	4.216D-01	-2.578D-02	3.689D+06	3.689D+06	2.482D-18	6.890D-01	0.0	2.781D-01	3.598D-07	0.0
11	4.674D-01	-2.794D-02	3.678D+06	3.678D+06	4.129D-18	6.956D-01	0.0	2.432D-01	7.889D-07	0.0
12	5.131D-01	5.225D-03	3.647D+06	3.647D+06	8.231D-18	7.055D-01	0.0	2.013D-01	1.761D-06	0.0
13	5.589D-01	-2.371D-02	3.633D+06	3.633D+06	3.660D-17	7.110D-01	0.0	2.013D-01	3.831D-06	0.0
14	6.045D-01	-4.047D-02	3.632D+06	3.632D+06	1.330D-16	7.145D-01	0.0	2.013D-01	8.096D-06	0.0
15	6.502D-01	-4.393D-03	3.643D+06	3.643D+06	2.046D-16	7.155D-01	0.0	1.525D-01	1.743D-05	0.0
16	6.960D-01	6.883D-03	3.647D+06	3.647D+06	5.140D-16	7.036D-01	0.0	1.525D-01	3.402D-05	0.0
17	7.417D-01	1.781D-02	3.629D+06	3.629D+06	1.751D-15	6.617D-01	0.0	9.675D-02	6.327D-05	0.0
18	7.874D-01	-5.883D-03	3.631D+06	3.631D+06	7.516D-16	6.627D-01	0.0	9.675D-02	1.094D-04	0.0
19	8.331D-01	3.731D-02	3.610D+06	3.610D+06	2.469D-15	6.604D-01	0.0	1.083D-01	1.939D-04	0.0
20	8.788D-01	6.983D-04	3.621D+06	3.621D+06	4.771D-15	6.547D-01	0.0	1.126D-01	2.922D-04	0.0
21	9.246D-01	1.067D-02	3.616D+06	3.616D+06	6.319D-15	6.234D-01	0.0	9.675D-02	4.050D-04	0.0
22	9.703D-01	1.591D-02	3.619D+06	3.619D+06	1.662D-14	5.273D-01	0.0	2.403D-01	5.162D-04	0.0
23	1.016D+00	3.536D-02	3.700D+06	3.700D+06	6.542D-16	6.799D-01	0.0	0.0	0.0	0.0

FUEL MASS IN ALL PINS	TOTAL REACTIVITY	FUEL REACTIVITY
0.86419492D+02	-0.10191D+01	-0.10191D+01

Figure 15.3.1: Regular PINACLE Output

Variable	Description
K	The number of the axial cell in the cavity.
ZZPI	Axial location of the lower boundary of cell k, measured from the bottom of the pin, m.
UFPI	Velocity of the molten fuel/fission gas-mixture in the cavity, at the axial location ZZPI, m/s.
PRCA	Total pressure in the cavity, Pa.
PRFIPI	Partial pressure of fission gas in the cavity, Pa.
PRFVPI	Partial fuel vapor pressure in the cavity, Pa.
FUVOFR	Molten fuel volume fraction in the cavity.
FUMESM	Mass of fuel molten during the current time step, kg.
FIMESM	Mass of free fission gas added to cell K of cavity during the current time step, kg.
FALRAT	Failure ratio used to determine the initiation of cladding failure and switching to PLUTO2 or LEVITATE models. Cladding failure occurs when FALRAT ≥ 1 in any axial node.
PRNACA	Partial pressure of sodium vapor in the cavity, Pa.

Finally, a last summary line prints the fuel mass in all the pins in a subassembly and the total reactivity introduced by the molten fuel relocation calculated by PINACLE. The reactivity effect refers to the whole SAS4A channel, i.e., accounts for the number of assemblies grouped together in a SAS4A channel.

15.3.3.2 Optional Output

An optional printer plot of the axial fuel distribution in a channel can be obtained by setting the input **IPNPLT** = 1. This optional output is illustrated in Figure 15.4.1.

This plot prints the linear fuel density in g/cm for each axial cell. Note that the cell numbers that appear on the left refer to the coolant channel mesh, *not* to the fuel pin mesh. The ordinate indicates the linear density at each axial location in g/cm/pin. It is generally labeled in increments of 0.5 g/cm. The symbol "T" indicates the total amount of fuel at each location, while the "P" indicates the amount of solid fuel at the same location. The difference between the "T" and "P" values is thus a measure of the amount of molten fuel present at the location. Finally, the original amount of fuel at each location is marked by a "0". At locations where the total amount of fuel present "T" is the same as the original amount of fuel "0", only the "T" appears.

15.4 Future Directions for Modeling Efforts.

15.4.1 Annular Molten Region

The current version of PINACLE models the formation of circular central cavity within the fuel pin and the subsequent fuel relocation inside this central cavity. Although the formation of such a central cavity is likely in U-F metal pins and oxide fuel pins, the situation is different for the U-Pu-Zr pins currently considered for the metal fuel core. In the U-Pu-Zr pins the material redistribution, particularly the Zr migration, leads to the formation of an annular Zr-depleted region, with a melting point significantly lower than the central and outer fuel regions. As the fuel pin temperature increases, the SAS4A calculations for U-Pu-Zr pins may indicate the formation of an annular molten cavity, due to the presence of this annular region with a low melting temperature.

It is thus necessary to develop a PINACLE capability to treat the formation of an annular cavity and to model the hydrodynamic fuel relocation in an annular geometry. Furthermore, the heat transfer model HTRVPN will have to be changed to accommodate the heat transfer calculations in the central solid region and annular molten

cavity. Once this capability is developed it will be possible to use the results calculated by the fuel redistribution modules, to obtain a more complete picture of the U-Pu-Zr fuel pin behavior during the accident.

15.4.2 New Moving Material Components

The current in-pin hydrodynamic model describes the relocation of one fuel component and two fission gas components, i.e., the free and dissolved fission gas. The specific characteristics of the metal-fuel pins will require the addition of several new material components, as outlined below.

U-Pu-Zr fuel pins subjected to irradiation have a fuel composition that is dependent on both the radial and axial location. The composition of the molten fuel in the pin cavity will thus also depend on the initial location of the molten material, as well as on the time dependent in-pin material relocation and cavity extension. In order to describe this situation two new fuel material components will be needed in PINACLE, describing the Pu and Zr mass distribution, while the existing fuel material will be used for the U fuel component. Only one homogenized U-Pu-Zr composition will be present in the molten cavity at each axial location, but this composition will vary with the axial location.

The metal-fuel porosity can be partially filled with small amounts of liquid sodium originating from the original bond sodium used to fill the fuel-cladding gap. As the fuel pin swells during irradiation and reaches the cladding, much of the sodium relocates in the space above the fuel column, but some of it might end up logged in the fuel porosity. The presence of this sodium is taken into account now only in the heat-transfer models, by using a modified solid fuel conductivity which takes into account the presence of the porosity partially filled with sodium. The sodium component, however, must be accounted for after the occurrence of fuel melting, which will require the addition of a moving Na component to the PINACLE hydrodynamic model. The presence of the liquid sodium in the molten cavity will directly affect the physical properties of the moving mixture. Although at the pressure levels prevailing in the bottled cavity the liquid sodium is not expected to have a major effect on the pressure, this situation can change after the pin failure when the cavity pressure drops rapidly and the sodium vapor pressure might play a significant role. Adding the liquid Na component to the PINACLE hydrodynamic model will allow the modeling of the prefailure in-pin axial sodium relocation and will provide the correct initial conditions for the postfailure LEVITATE calculations.

A characteristic phenomenon in metal-fuel pins is the formation of a molten eutectic layer at the fuel cladding interface. Iron diffuses inwards into the fuel, leading to the formation of an alloy with a lower solidus temperature than the original fuel and thus to fuel liquefaction. This molten eutectic region progresses radially at the same time as the central molten fuel cavity. When the molten fuel region reaches the molten eutectic region the two molten components can mix and axial relocation of the molten eutectic is possible. This situation is not modeled currently. Although the formation of the cladding molten eutectic region is taken into account in the DEFORM-5 cladding failure calculations, the iron present in the molten fuel or cladding cannot relocate axially or mix with the molten fuel. A moving cladding material component will be needed in the PINACLE hydrodynamic model in order to model the molten fuel and cladding mixing and axial relocation. The fuel and cladding components interact chemically, resulting in a mixed molten alloy. The mixing and interaction of the fuel and cladding directly affects the physical properties of the molten moving mixture. The axial relocation of the molten cladding material itself is expected to have a small reactivity effect prior to cladding failure but, more importantly, it will provide the correct initial conditions for the postfailure fuel relocation, when rapid in-pin motion can lead to more significant reactivity effects due to in-pin cladding relocation.

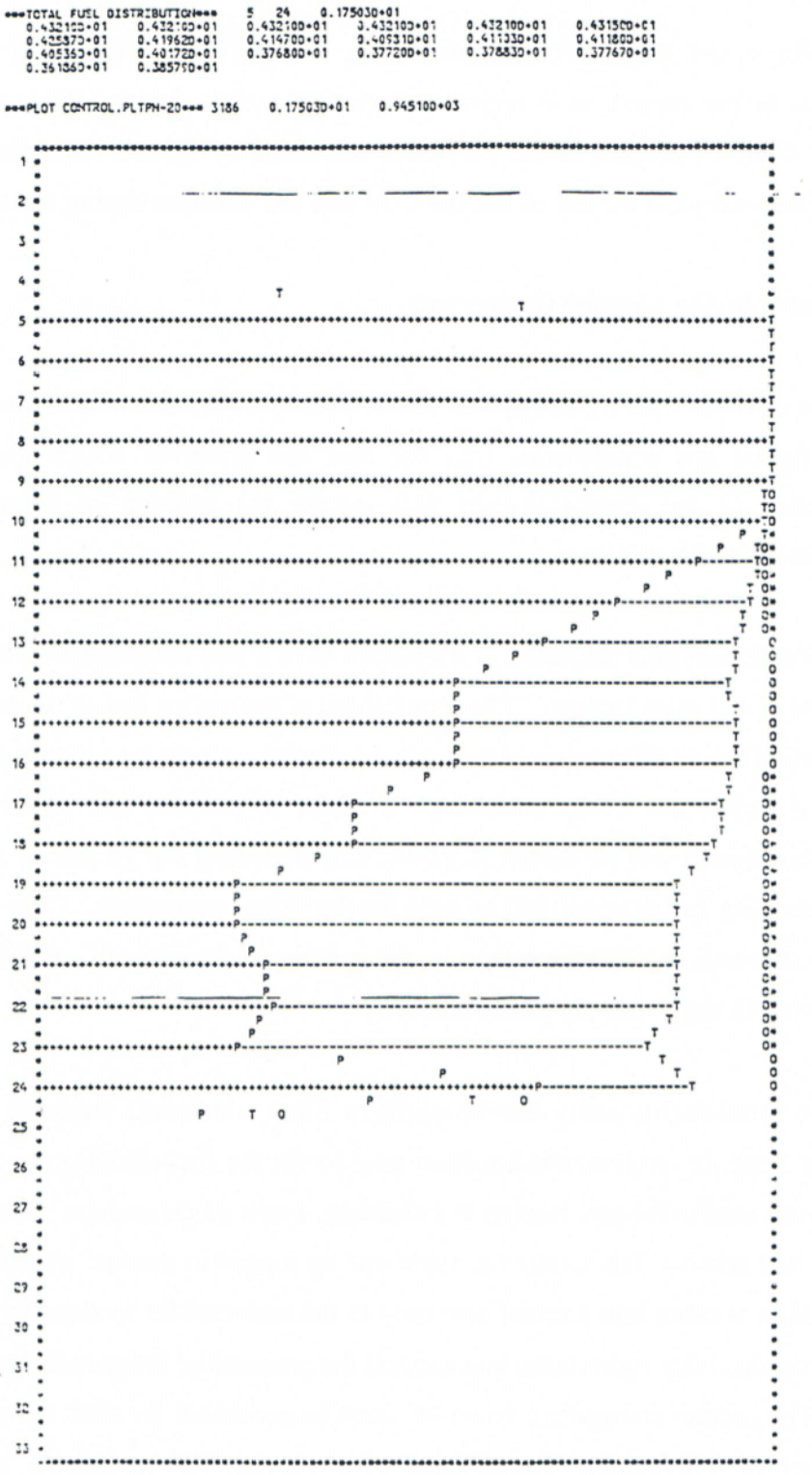


Figure 15.4.1: Optional PINACLE Output

15.4.3 Inner Cladding Ablation

In metal-fuel pins the molten material region can extend radially past the original inner cladding interface as the molten fuel region is connected to the molten eutectic region. At present, the molten cavity can extend only to the initial inner cladding interface and cladding ablation can occur only at the outer cladding interface, after the initiation of the postfailure fuel motion model LEVITATE. When the molten cladding moving component is implemented, a model describing the cladding ablation at the fuel-cladding interface will be necessary. This model will allow the radius to increase past the original inner cladding radius and will allow the mixing of the molten eutectic with the moving components in the pin cavity as needed.

15.4.4 Composition-Dependent Moving Mixture Properties

The physical properties of the molten mixture in the cavity are dependent on the material composition, i.e., the proportion of U, Pu, Zr and Fe in the mixture. The physical properties affected by the composition include the solidus and liquidus temperatures, the conductivity, specific heat and density. Composition dependent functions representing these physical properties will have to be implemented in the PINACLE routines.

15.4.5 Fuel Freezing in the Plenum

The present model describes the radial heat-transfer between the fuel ejected above the fuel-pin and the cladding. The heat-transfer model will be expanded to describe the axial heat transfer between the molten fuel and the liquid sodium slug. A fuel freezing model describing the formation of a fuel crust in the gas plenum is already available in PINACLE. However, the accuracy of this model is limited by the fact that only one axial fuel cell, of variable length, is allowed above the molten fuel-pin. This limitation was introduced mainly because of the limited number of axial locations (24) available in SAS4A for the fuel-pin arrays. In the future the models describing the molten, heat-transfer and freezing of the ejected fuel will be expanded by allowing the fuel region to cover multiple axial cells located above the original fuel pin. A precondition for this development is an increase in the size of the SAS4A fuel-pin arrays.

15.4.6 Initiation of Axial Motion

The results of the TREAT experiment analyses performed with PINACLE, such as TS-2, M2 and M3 series indicate that the temperature of the fuel pin top interface is an important element for the prediction of the time of onset of axial fuel ejection. However, it is likely that the pressure difference between the molten fuel cavity and the upper gas plenum also plays a role in the initiation of axial fuel ejection. The model describing the onset of fuel motion will be enhanced in the future to incorporate both the interface temperature and pressure difference effects.

The timing of the in-pin molten fuel relocation initiation plays an important role in determining the accident sequence. If the in-pin fuel motion is initiated prior to cladding failure, a significant amount of negative reactivity is added at a high rate, causing a rapid decrease in power and reactivity. Thus, if cladding failure occurs later, it is likely to occur at considerably lower power and reactivity levels. The timing of the rapid in-pin fuel relocation initiation is determined by the breach of the solid fuel at the top of the pin, which separates the molten pressurized cavity from the gas plenum, usually at a lower pressure. This event is influenced both the temperature distribution near the top of the fuel pin and by the pressure difference between the molten cavity and the gas plenum.

As present the fuel temperatures in the top axial fuel cell are used to determine the timing of the onset of rapid in-pin fuel relocation. Only a radial fuel distribution is calculated and any axial heat transfer is neglected. However more detailed two-dimensional calculations indicate that near the top of the fuel pin the axial heat transfer between the fuel and the molten sodium present above the fuel pin becomes an important element in determining the temperature distribution. Because the top of the solid fuel pin actually controls the onset of

rapid in-pin molten fuel relocation, it is necessary to implement at two-dimensional temperature calculation in the top axial fuel cell.

It is likely that the pressure difference between the molten cavity and the upper gas plenum also plays a role in the initiation of the axial fuel upward ejection. The current models do not take into account this pressure difference in the initiation of in-pin rapid fuel relocation, although the pressure difference is considered in the hydrodynamic models and plays an important role in determining the rate of fuel ejection after the onset of in-pin fuel relocation. The model describing the onset of rapid molten fuel motion will be enhanced in the future to incorporate the pressure difference effects in addition to the fuel temperature distribution.

15.4.7 Fuel Blanket and Sodium Slug Motion

A separate momentum equation describing the motion of the fuel blanket stack and/or liquid sodium slug will be added. Currently, this stack is modeled as moving together with the molten fuel until it reaches a rigid obstacle. Afterwards, the stack is assumed immobile and the fuel ejection is governed by the pressure difference between the upper cavity cell and the cell above the fuel pin. Although the current treatment is physically justified and necessary, for numerical stability reasons, when the amount of molten fuel ejected is small, the addition of a separate momentum equation for the pellet stack and/or liquid sodium slug will significantly increase the flexibility of the model when calculating the ejection of larger amounts of molten fuel above the active fuel column.

At present, the calculations describing the transient temperatures of the liquid sodium slug and fission gas plenum continue after the initiation of the rapid in-pin fuel relocation, ignoring the boundary heat-transfer changes due to the sodium slug relocation.

In fact the sodium slug is moving upwards, and thus the surrounding cladding temperatures change, leading to different heat fluxes. In addition, the axial heat transfer occurring at the lower slug boundary between the liquid sodium and molten fuel is likely to be significant. In the case of the fission gas, the upward motion of the sodium slug causes a decrease of the fission gas volume and boundary area. While the volume decrease is now accounted for in the pressure calculations, the heat-transfer model must be modified to account for the changes in the cladding area in contact with the plenum gas.

15.4.8 PINACLE Termination Upon Fuel Freezing and Restart Upon Remelting

For the examination of transients with decreasing power levels after the initiation of PINACLE, the addition of the capability of modeling fuel freezing in the pin cavity will also be necessary. In order to allow the analysis of long transients even after PINACLE has been initiated, it is necessary to develop the capability of discontinuing the PINACLE calculations whenever the fuel in the pin cavity freezes again and restarting the PINACLE calculations at a later time if necessary. This will allow the SAS4A calculations to proceed with significantly larger time steps.

15.5 References

15-1. A. M. Tentner et al., "The SAS4A LMFBR Whole Core Accident Analysis Code," *International Topical Meeting on Fast Reactor Safety*, Knoxville, TN, April 1985.

15-2. A. M. Tentner and H. U. Wider, "LEVITATE-A Mechanistic Model for the Analysis of Fuel and Cladding Dynamics Under LOF Conditions," *Intl. Mtg. on Fast Reactor Safety*, Seattle, WA, 1979.

15-3. H. U. Wider, "PLUTO2 - A Computer Code for the Analysis of Overpower Accidents in LMFBRs," *Trans. Am. Nucl. Soc.*, vol. 27, p. 533, Nov. 1977.

15-4. A. M. Tentner and D. J. Hill, "PINACLE - A Model of the Pre-Failure In-Pin Molten Fuel Relocations," *Trans. Am. Nucl. Soc.*, Boston, MA, June 1985.

15-5. Argonne National Laboratory, Unpublished information, 1975.

15-6. R. B. Bird, W. E. Stewart, and E. N. Lightfoot, *Transport Phenomena*, John Wiley & Sons, Inc., New York, 1960.

15-7. R. G. Deissler, "Analysis of Turbulent Heat Transfer, Mass Transfer, and Friction in Smooth Tubes at High Prandtl and Schmidt Numbers," NACA Report 1210, 1955.

15-8. R. E. Henry, M. A. Grolmes, and H. K. Fauske, "Pressure Pulse Propagation in Two-Phase One- and Two-Component Mixtures," ANL-7792, Argonne National Laboratory, Argonne, Illinois, March 1971.

15-9. W. R. Robinson et al., Unpublished information, Argonne National Laboratory, 1985.

15.6 Nomenclature

Symbols	Description	Units
Superscripts		
l	Linear	
n	Time step n	
‘	Prime always indicates that a quantity refers to a unit of generalized smear volume	
Subscripts		
ac	Acoustic	
bk	Lower boundary of node K	
ca	Cavity	
cl	Cladding	
fica	Free fission gas in molten pin cavity	
fr	Friction	
fsca	Dissolved fission gas in the molten pin cavity	
fuca	Mobile fuel in the molten pin cavity	
fufi	Fuel and fission gas	
fvca	Fuel vapor in the molten pin cavity	
i	Axial coolant channel index, specific enthalpy	
if	Interface	
K	Axial pin or cavity index	
liq	Liquidus	
me	Melting, melt-in	
min	Minimum	

continues on next page

Table 15.6.1 – continued from previous page

Symbols	Description	Units
max	Maximum	
Na	Sodium	
Nl	Liquid sodium	
Nv	Sodium vapor	
pin,pi	Pin	
rl	Dissolved fission-gas release	
S’	Sink or source per unit of generalized smear volume	
sol	Solidus	
un	Uncompressed	
vg	Vapor and gas	
vi	Viscous	
z	Axial location z	
Greek		
α	Void fraction	
β	Delayed neutron fraction	
γ	C_p/C:sub:*v*	
δ	Partial derivative	
Δ	Derivative in finite differences	
κ	Compressibility	1/Pa
π	Circumference of a circle divided by its diameter	
ρ	Theoretical density	kg/m^3
ρ’	Generalized linear density	kg/m^3
σ	Surface tension	J/m^2
θ	Generalized volume fraction	
Regular		
A	Cross section area	m^2
A’	Interaction or surface area per unit of generalized smear volume	1/m
AXMX	Input, reference cross section area; recommended input value is subassembly cross section area	m^2
BFR	Input, exponent of liquid friction coefficient	
BFRV	Input, exponent of vapor friction coefficient	
C	Specific heat	J(kg·K)
CDFU	Input, fuel conductivity	J(m·s·K)
CDNL	Input, liquid sodium conductivity	J(m·s·K)
CDVG	Input, conductivity of sodium vapor	J(m·s·K)
CFCOFV	Input, fuel vapor condensation coefficient	$J(m^2 \cdot s \cdot K)$
CIA3	Input, constant in the Deissler heat-transfer-correlation	
CIFRFU	Input part of the fuel friction calculation	
CIREFU	Input, Reynolds number above which fully turbulent fuel flow is assumed for friction calculation	
CIRTFS	Input, controls dissolved gas release	1/s
CMFU	Input, liquid fuel compressibility	1/Pa
CPFU	Input, fuel heat capacity	J/(kg·K)
D	Diameter, hydraulic diameter	m
e	Internal energy	J/kg
EGBBLY	Input, internal fuel energy below which fuel freezing begins	J/kg
EGFULQ	Input, internal fuel energy at the liquidus	J/kg
EGFUSO	Input, internal fuel energy at the solidus	J/kg

continues on next page

Table 15.6.1 – continued from previous page

Symbols	Description	Units
EPCH	Input, controls use of advanced pressure in in-pin calculation	
F	Friction factor	
FNFUAN	Input, controls when annular fuel flow in molten pin cavity is assumed	
FNMECA	Fraction of node width of radial node on cavity boundaries which melts in per PINACLE time step	
FNMELT	Input, controls location of cavity boundary	
FNPOE	Relative power level	
FUELMS	Initial fuel mass in radial fuel-pin node	kg
FUMASS	Initial total fuel mass in an axial fuel node	kg
FUMS	Current fuel mass in the radial fuel pin node on the cavity boundary	kg
g	Gravity	m/s^2
GAMSS	Input, fraction of power going into direct heating of structure	
GAMTNC	Input, fraction of power going into direct heating of coolant	
GAMTNE	Input, fraction of power going in to direct heating of cladding	
h	Heat-transfer coefficient	$J/(m^2 \cdot s \cdot K)$
H	Heat-transfer coefficient times heat-transfer area	J/(s·K)
I	Axial node index in coolant channel	
IDIFF	Offset between pin and channel grid. The first pin node is at the same elevation as channel node IDIFF + 1	
k	Conductivity	J(m·s·K)
K	Axial index in the pin	
KKMX	Uppermost node in the molten pin cavity	
KZPIN	Coolant channel zone which contains the fuel pins	
KK1	Lowermost node in the molten pin cavity	
L	Length	m
NRPI	Number of pins per subassembly	
NT	Number of radial fuel pin nodes	
Nu	Nusselt number	
P	Pressure	Pa
PORFR	Porosity fraction	
POW	Input, power in highest rated axial fuel pin node	W
POWCOF	Exponent in the exponential function which gives the power history during a neutron's time step	
Pr	Prandtl number	
PSHAPE	Input, axial pin power shape	
PSHAPR	Input, radial power shape in a pin	
Q	Fission heat source	
r	Radius	W/kg
R	Gas constant or radius	J/(kg·K)
RETFG2	Total fission gas mass in the radial fuel pin node at the cavity boundary before it began melting into the cavity	kg
Re	Reynold's number	
RGAS	Input, gas constant for fission gas	J(kg·K)
S	Mass sink or source	kg/s
S'	Mass sink or source per unit of generalized smear volume	kg/(m·s)
St	Stanton number	
t	Time	S

continues on next page

Table 15.6.1 – continued from previous page

Symbols	Description	Units
T	Temperature	K
u	Velocity	m/s
VIFG	Input, viscosity of fission gas	kg/(m·s)
VIFULQ	Input, viscosity of liquid fuel	kg/(m·s)
VOLUME	Volume	m^3
z	Axial coordinate	m

CHAPTER

SIXTEEN

LEVITATE: VOIDED FUEL MOTION

16.1 Overview

16.1.1 Historical Background and Physical Description of the LEVITATE Model

In an unprotected loss-of-flow (LOF) accident in an LMFBR, the boiling and voiding of the coolant sodium in the high-power subassemblies will lead to an overpower situation if the sodium void worth of the reactor considered is moderately positive. A high-power level causes the subsequent events in the voided channels such as cladding motion, fuel-pin breakup, and fuel motion to occur nearly simultaneously. A positive reactivity contribution from these effects could lead to a potentially energetic LOF-driven-TOP accident, whereas a negative contribution would most likely lead the accident into a transition phase much like that predicted as the likely outcome of a LOF accident in a low-void-worth LMFBR.

The LEVITATE model [16-1] has been designed to treat both the high-power and the near-nominal power conditions in voided assemblies. This means that cladding and fuel motion can be treated in a combined or sequential fashion. The earlier CLAZAS [16-2] and SLUMPY [16-3] models in SAS3A [16-4] were designed only to treat these phenomena in a sequential fashion. The new LEVITATE model also treats several relevant phenomena not considered in the earlier models. The most important of these are several pin-disruption modes, continuous fuel-steel flow regimes and fuel-steel crust and plug formation, and a tight coupling with the sodium dynamics. LEVITATE has also been designed to incorporate a fuel-chunk model, describing the motion of the solid fuel chunks present in the coolant channel. This model has become operational in the developmental version of LEVITATE and was not available in the initial release version of SAS4A. Since two-phase sodium which is generated by the chugging of the lower sodium slug may penetrate the disrupted region, fuel may be pushed upwards or "levitated," prompting the name of this model.

The LEVITATE development used the PLUTO2 code [16-5] as a starting point and still shares some features with that code. PLUTO2 and LEVITATE are complementary models with some degree of overlap. PLUTO2 can treat overpower situations in sodium-filled channels, i.e., fuel-coolant interactions and fuel sweepout, but not the later cladding motion, fuel-steel mixing, and pin disintegration.

16.1.2 Physical Description of the LEVITATE Model

LEVITATE models the fuel subassembly in a one-dimensional geometry, assuming that all pins in the subassembly behave coherently. Three basic thermal-hydraulic models are used for each subassembly:

1. The hydrodynamic model describing the cavities inside the fuel pins, which initially contain liquid fuel and fission gas.
2. The hydrodynamic model describing the coolant channel, bounded by the outside cladding surface and the hexcan wall. This channel contains initially liquid sodium and sodium vapor.
3. The heat-transfer and melting/freezing model, describing the solid fuel-pin stubs, which separate the outer channel from the inner cavity.

A typical LEVITATE configuration illustrating some of the recently introduced models is presented in Figure 16.1.1. Other features presented in this figure are introduced below.

16.1.2.1 Fuel-pin Disruption and In-pin Fuel Motion

As the loss-of-flow accident proceeds, the inside of the fuel pin begins to melt, leading to the formation of an internal cavity. This cavity is filled with a mixture of molten fuel and fission gas, and expands continuously, both radially and axially, due to fuel melting. The fuel-gas mixture in the cavity is pressurized due to the presence of fission gas and fuel vapor. As the cavity walls continue to melt and the cladding temperature approaches the melting point, this continued pressurization of the cavity leads to fuel-pin failure, as illustrated in Figure 16.1.2. Based on the mechanism of fuel-pin failure, two disruption modes are currently modeled in LEVITATE.

1. *Total disruption of the fuel pin at a certain axial location.* When a large fraction of the pin has become molten at a certain axial location and the cladding is no longer effective in restricting radial motion, the fuel pin is totally disrupted. In this case, the area previously occupied by pins becomes part of the coolant channel and only the two fuel-pin stubs remain, as shown in Figure 16.1.1. The flow of molten fuel and fission gas inside the stubs continues to be described by a hydrodynamic model. As exemplified in Figure 16.1.1, this mode of disruption leads to significant area changes in the coolant channel. This situation has made necessary an accurate treatment of abrupt area changes in the hydrodynamic modeling.
2. *Mechanical cladding failures at a certain axial location.* If a large fraction of the fuel is still in a solid state, the initial geometry of the fuel pin remains intact. However, due to low cladding resistance and a higher pressure inside the pin than in the coolant channel, molten fuel and fission gas from the pin cavity are ejected into the coolant channel through a cladding rupture.

Due to the fuel-pin failure, the inner cavity is connected to the coolant channel which is at a significantly lower pressure, and the molten fuel inside the pin is accelerated rapidly toward the pin failure location. This motion is modeled by the in-pin hydrodynamic model. An ejection model transfers molten fuel and fission gas from the pin cavity to the coolant channel, thus connecting the two main hydrodynamic models.

16.1.2.2 Hydrodynamics of the Coolant Channel

Before the molten fuel-fission gas mixture is ejected from the pin cavities, the coolant channel contains only sodium and perhaps molten steel; but as the fuel and fission gas begin to interact with these original components, a very complex situation develops, involving a large number of components that have to be tracked separately. The moving components in the channel are solid and liquid fuel, solid and liquid steel, fission gas, and vapors of fuel, steel and sodium. The material motion is described by a multi-component, multi-phase, nonequilibrium hydrodynamic model. The region described by this model is bounded axially by the liquid sodium slugs, and is generally referred to as “the interaction region”. This region can increase or decrease, depending on the dynamics of the liquid slugs which are described by a simple incompressible model. The dependent variables

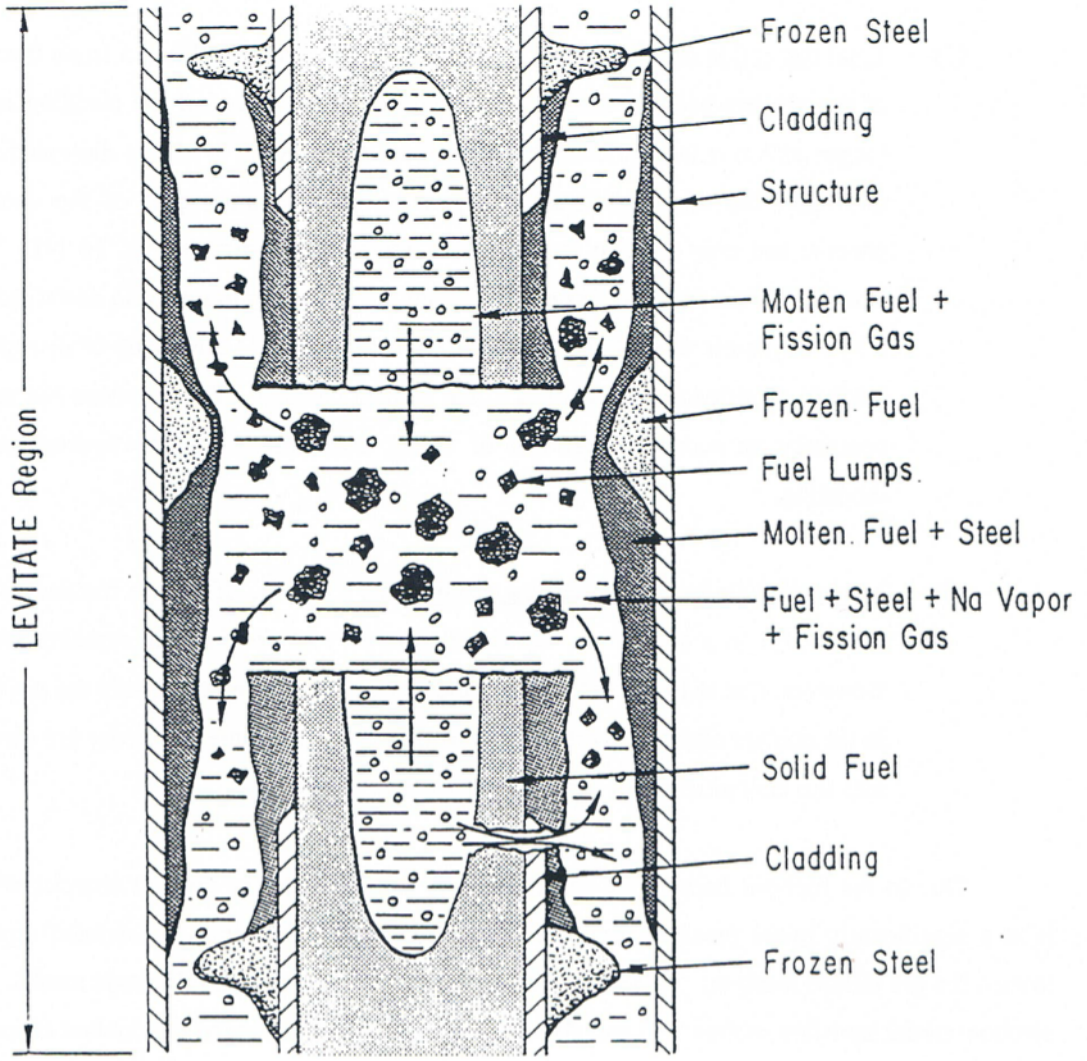


Figure 16.1.1: Typical LEVITATE Configurations Fuel-pin Cavity Formation

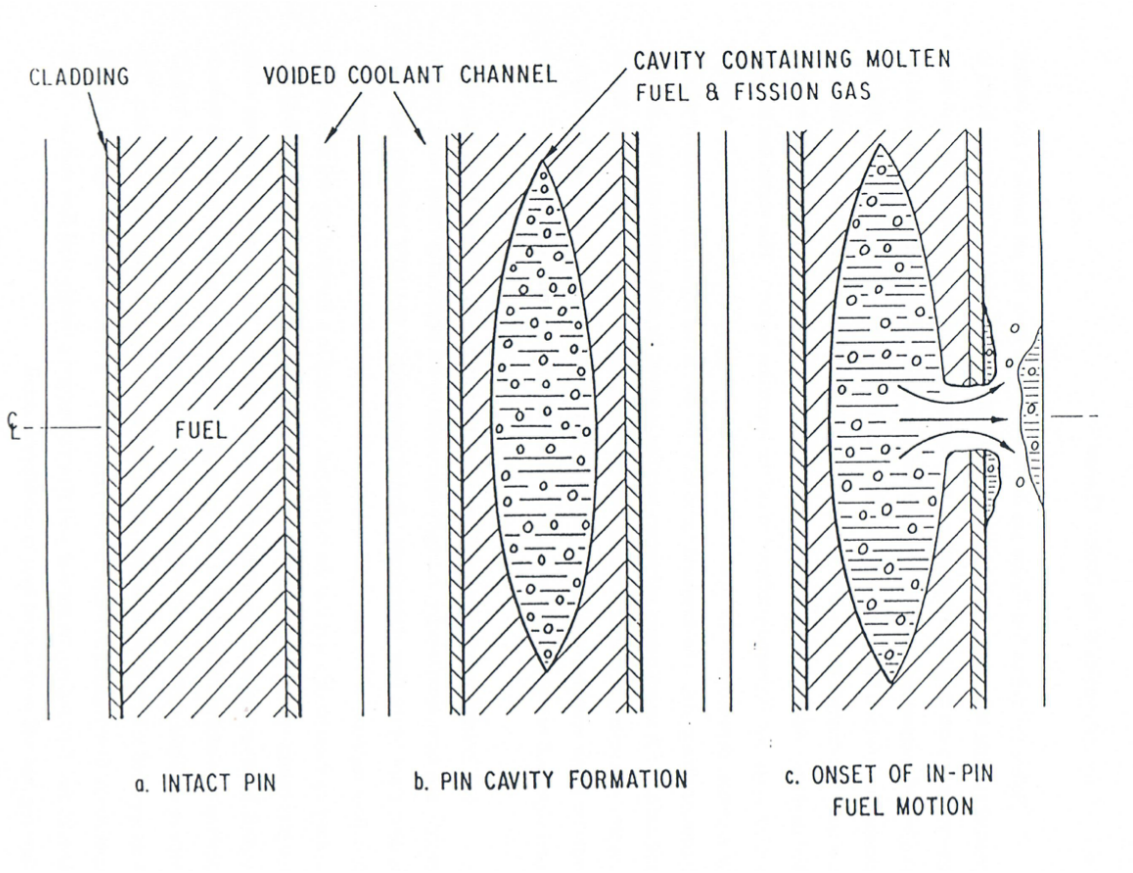


Figure 16.1.2: Fuel Pin Cavity Formation

in the interaction region are the density, velocity and enthalpy. A separate mass and energy equation is solved for each component, but only three coupled momentum equations for three velocity fields are solved. The components treated together in the velocity fields are: (a) liquid fuel and liquid steel, (b) fission gas, fuel vapor, steel vapor and two-phase sodium and (c) solid fuel chunks and solid steel chunks.

The interaction between the different components present in the channel, i.e., mass, energy, and momentum transfer, is largely determined by the local configuration which, in turn, is determined by the flow regime used. Earlier codes such as SLUMPY used only particulate fuel flow regimes, which may lead to unrealistically rapid fuel dispersal. The assumption underlying these models was that the molten fuel contained in a disrupted pin cell ejected from the pins breaks up into droplets upon entering the coolant channels. Tentner et al. [16-6] have argued, however, that such a particulate flow cannot be justified in sodium-voided regions which develop soon after pin failure. First, there is no apparent reason why molten fuel which contacts little or no liquid sodium should fragment. Second, most of the frozen fuel found in TREAT tests appeared to be in continuous form, rather than in the form of rounded frozen droplets. The continuous flow regimes modeled in LEVITATE are: a bubbly fuel flow regime, a partial annular fuel flow regime, an annular steel flow regime and a bubbly steel flow regime. These flow regimes will be described later in considerable detail.

16.1.2.3 The Freezing/Melting Models Describing the Solid Fuel-pin Stubs

At the axial locations where solid fuel pins are still present, the coolant channel is separated from the pin cavity by the cladding and the remaining solid fuel. The temperature field in the cladding and fuel is calculated by a transient heat-transfer model, using the temperatures in the channel and cavity as boundary conditions. Continuous melting occurs at the fuel pin cavity boundary, leading to an increase in cavity diameter and addition of molten fuel and fission gas to the moving components in the cavity. The situation is more complicated at the channel boundary. It has been previously argued that, in the sodium-voided regions which develop during a loss-of-flow situation, continuous fuel flow regimes are likely to exist. Under these conditions, the molten-fuel/cladding interface temperature usually falls between the freezing temperatures of these substances, resulting in solidification of the initially molten ceramic fuel and melting of the initially solid steel. The assumption which was generally made in the modeling of the simultaneous fuel freezing steel melting phenomena was that the frozen fuel crust is mechanically stable and does not break up under the influence of fluid frictional shear or buoyancy forces [16-7]. However, experiments conducted by Spencer et al. [16-8] indicate that this is not the case for the flow of molten fuel in pin bundles. In these experiments, significant steel ablation and fuel-steel mixing were observed that could not have occurred in the presence of a stable crust [16-9].

The fuel-freezing model used in LEVITATE allows for the formation of a partial fuel crust when the temperature of the fuel in the channel drops below an input freezing temperature. This input temperature is always between the liquidus and solidus temperatures.

When the dominant component in the channel is molten steel, steel freezing can occur, leading to the formation of steel plugs.

The temperature of the fuel crust, at any given location, is calculated by the heat-transfer model. Depending on its temperature and other local conditions, which will be described in detail later, the fuel crust can continue to grow, can start to remelt or can break up when the underlying cladding begins to melt.

16.1.2.4 Geometry Description and Interaction among LEVITATE Models

The geometry modeled in LEVITATE is illustrated schematically in Figure 16.1.3, which will also be used to describe the interaction of the physical models described previously.

LEVITATE calculates all the thermal hydraulic events that occur in a pin subassembly. The subassembly is bounded axially by the lower and upper plena. Although only one pin is shown in Figure 16.1.3, LEVITATE will account for the appropriate number of pins per subassembly, as specified in the input description. The hydrodynamic in-pin calculations are performed on a mesh grid using the subscript K, having the origin at the bottom of the lower blanket. The top node of the upper blanket is indicated by the variable MZ. The active fuel core extends from KCORE1 to KCORE2. The fuel pin cavity, which increases gradually both radially and axially, cannot extend beyond the active core boundaries. The disrupted pin region extends from KDISBT to KDISTP. The hydrodynamic coolant channel calculations as well as the freezing and melting calculations are performed on a mesh-grid using the subscript I, with the origin at the bottom of the fuel pins. The integer IDIFF indicates the offset between the I and K grids, i.e.,

$$I = K + \text{IDIFF} \tag{16.1-1}$$

The top node in the subassembly and in the LEVITATE domain is indicated by the variable IITP. The hydrodynamic and melting/freezing models, however, operate only in the interaction region, which extends from IFMIBT to IFMITP. This region is bounded axially by the lower and upper sodium slugs and can expand or contract depending on the dynamics of the slugs. The slug motion is determined by the pressure difference acting on them, e.g., the lower slug motion is determined by the pressure difference between the cell IFMIBT and the lower plenum. The slug motion is calculated by an incompressible model to be described later. Each material component is restricted to its own region, with moving boundaries which are tracked continuously. The procedure reduces significantly the undesirable numerical diffusion effects. Figure 16.1.3 illustrates only the liquid fuel region, which extends from the cell IFFUBT to IFFUTP. The pin disrupted region extends from the cell IDISBT to IDISTP, corresponding to the integers KDISBT and KDISTP on the K grid. The cladding rupture extends from IFRIBT to IDISBT-1 and from IDISTP+1 to IFRITP. If no pin disruption is present, the cladding rupture extends from IFRIBT to IFRITP. It is noted that IFRIBT and IFRITP are the lowermost and uppermost cells with a cladding rupture but that it is not necessary that all the intermediate cells exhibit a cladding rupture, although this is generally the case. The in-pin hydrodynamic model is connected to the coolant channel hydrodynamic model via the ejection process. Fuel and fission gas can be ejected from the cavity into the channel either via the cladding rupture or, when the pins have been disrupted, via the open ends of the remaining pin stubs. These ejection processes are described by appropriate physical models.

16.1.3 Interaction of LEVITATE with Other Models within the SAS4A System

16.1.3.1 LEVITATE Initiation

LEVITATE can be initiated by only two routines of the SAS4A code, i.e., FUINIT and PLUTO2.

The initiation of fuel motion is decided, for any given channel, in the routine FAILUR, called from TSTHRM. If FAILUR predicts the onset of fuel motion, the module FUINIT is called to prepare the transition to the fuel motion modules, LEVITATE or PLUTO2, and to decide which of these two models should be used (Figure 14.7.1 in Chapter 14). Several conditions characteristic of LOF situations have to be met for the initiation of LEVITATE:

- Pin failure must occur after sodium boiling has occurred.
- Only one large sodium vapor bubble must be present in the coolant channel. The bubble must extend over at least four axial cells.

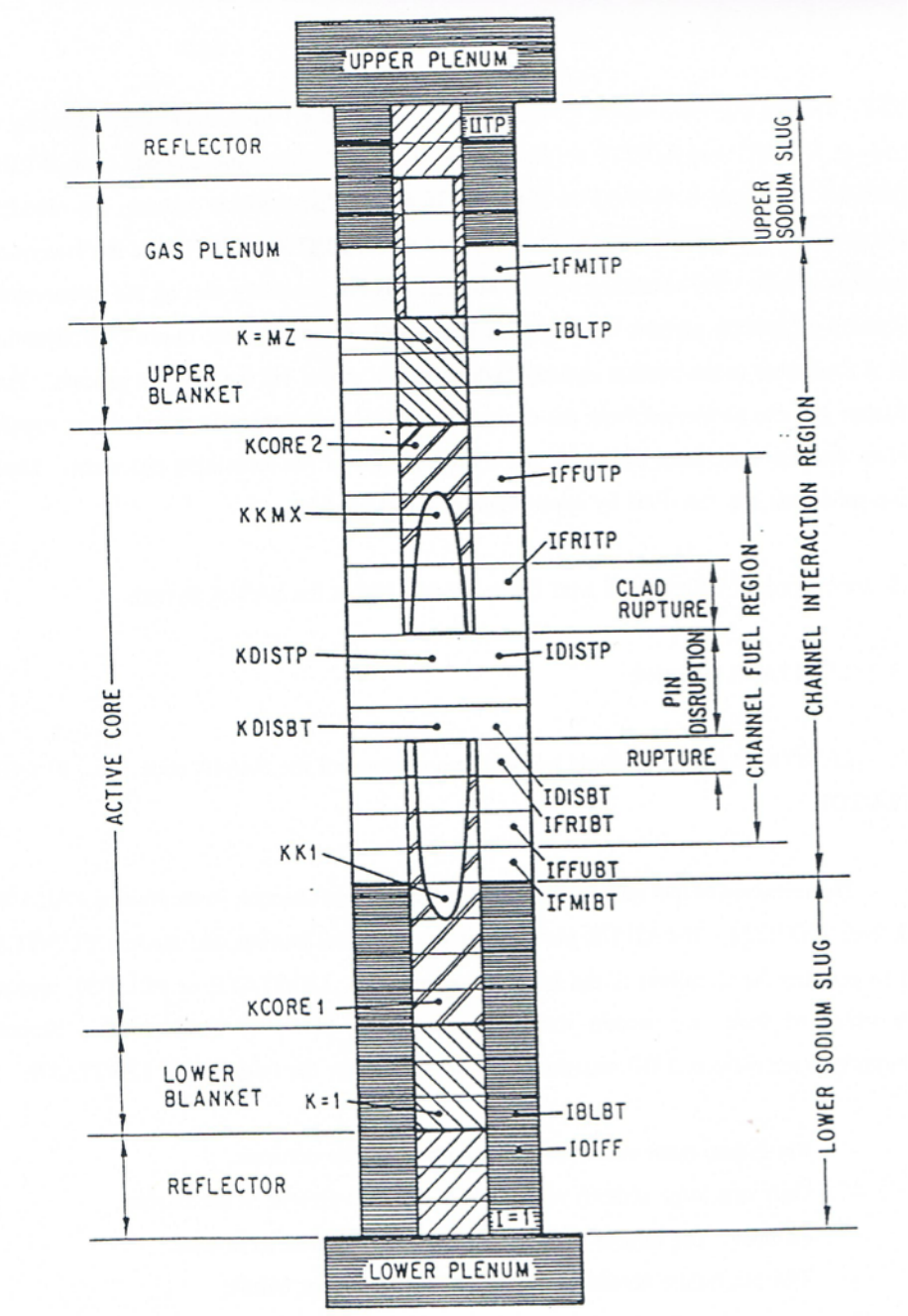


Figure 16.1.3: Geometry Modeled by LEVITATE

- The pin failure should be located within the vapor bubble.

The objective of these constraints is to insure that the pin failure occurs in a largely voided region, where the LEVITATE models are valid. If, however, the pin failure occurs in a region containing significant amounts of liquid sodium, PLUTO2 should be initiated. In such a case, the molten fuel is likely to fragment into droplets upon contact with the sodium, leading to the more rapid fuel sweepout characteristic of TOP situations. This type of event is modeled in PLUTO2, but not in LEVITATE. If any of the above conditions are not met, the PLUTO2 fuel motion model will be initiated by FUINIT. One particular case must be noted. If two or more bubbles are present, a check is made for the presence of a dominant voided region. If such a region is found (length of voided region/length of boiling region >0.7), a flag is set (ILEPLI = 1), which will then be used in PLUTO2, as described below. Control is still transferred to PLUT02, whether or not a dominant bubble has been found.

When LEVITATE is initiated in FUINIT, an interface routine, LESAIN, is called to prepare all variables characteristic for LEVITATE. If the steel motion module, CLAP, was operational prior to LEVITATE initiation, an additional interface routine, LECLIN, is called from FUINIT. LECLIN is used to translate the CLAP steel-related variables to LEVITATE variables. Then two initialization routines, common for LEVITATE and PLUTO2, are called: PLINPT and PLSET. At the same time the flag ICALC used in TSTHRM is set to 2 [ICALC(ICH)=2] the value reserved for LEVITATE. Control is then returned to TSTHRM, which checks on ICALC(ICH) to decide which model has to be used in channel ICH.

The other path for LEVITATE initiation is via PLUTO2. Although some accident sequences can begin as TOP situations in any given channel, eventually the presence of the molten fuel in the coolant channel leads to a disrupted geometry, requiring the use of LEVITATE. The PLUTO2 module checks at the end of each primary time step whether or not control should be transferred to LEVITATE. Control should be transferred to LEVITATE whenever the original bundle geometry has to be changed due to steel ablation or possible pin disruption. Control is also transferred to LEVITATE whenever the fuel vapor pressure in the channel becomes significant. PLUTO2 does not model geometry changes due to ablation or pin disruption, and the fuel vapor is not currently included in the PLUTO2 channel hydrodynamic model.

The criteria for the transfer are as follows:

- Three or more axial cladding cells (see the input variable NCPLEV) are completely molten, indicating that steel ablation and fuel-steel mixing should occur, or
- The temperature of the molten fuel is above 4000 K, indicating the presence of a significant fuel vapor pressure, or
- The flag ILEPLI is found to have the value 1, indicating the presence of an initial large voided region.

When any of these criteria is satisfied, the flag ICALC(ICH) is set to 2, and a number of LEVITATE variables not used in PLUTO2 are set by executing LEPLIN. Control is then returned from PLUTO2 to TSTHRM, which will transfer control to LEVITATE. The relationship between LEVITATE and other SAS4A modules is illustrated schematically in Figure 16.1.4.

16.1.3.2 LEVITATE Calculations

Once the LEVITATE initialization routines have been executed and the flag ICALC (ICH) has been set to 2, the SAS4A transient driver TSTHRM will transfer control to the LEVITATE driver routine LEVDRV. LEVDRV will retain control and advance the solution in the channel ICH until the end of the primary loop time step is reached. It should be noted that once LEVITATE is initiated, the coolant time step is set equal to the primary-loop time step and these two steps can be used interchangeably. The flow chart in Figure 16.1.5 shows the logic of the LEVITATE driver.

First LEVDRV will execute LESET2. This subroutine initializes all temporary integers and arrays. These are values that can be calculated using the permanent quantities. They are kept only as long as LEVDRV retains

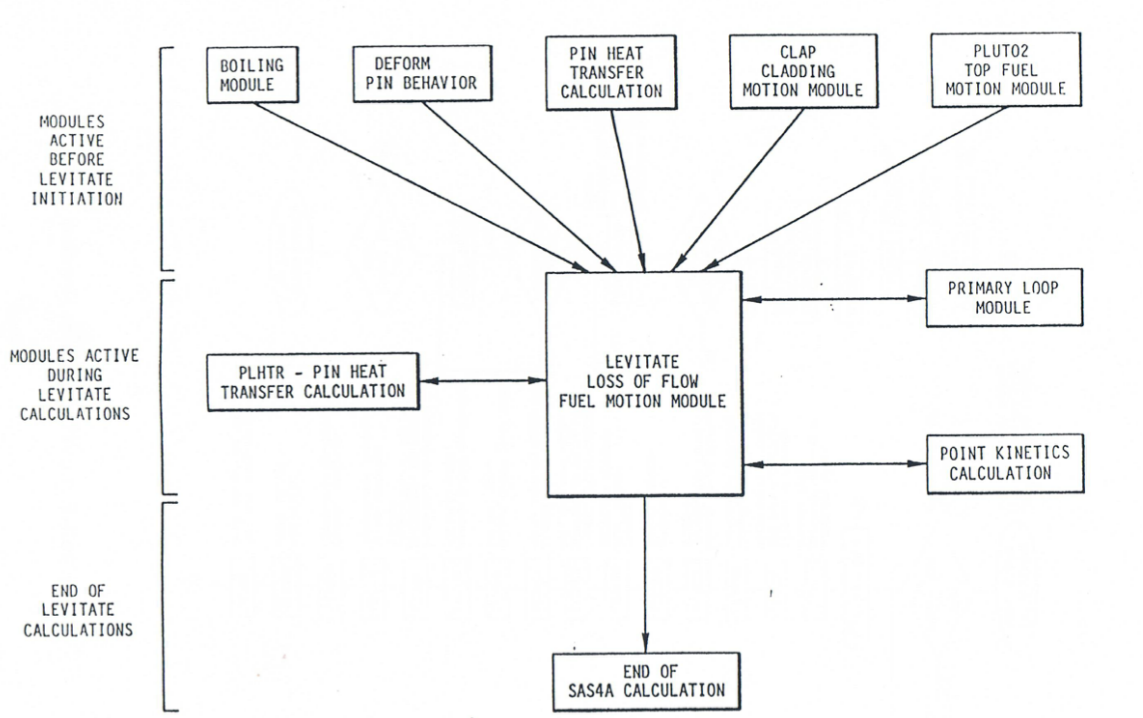


Figure 16.1.4: Relationship between LEVITATE and Other SAS4A Modules

control in the channel ICH. The solution is then advanced in the channel ICH by calling a sequence of subroutines from LEIF through LEDISR. The solution for the hydrodynamic in-pin model and the ejection of material from the pin cavities into the coolant channel are obtained in the routines LElPIN and LE2PIN. All the other routines in the sequence mentioned above are used to model the thermal and hydrodynamic processes which occur in the coolant channel.

Next, the LEVITATE driver routine determines the maximum time step acceptable for the coolant channel calculation in the next cycle. This value is compared with the maximum time step calculated for the in-pin model in LE2PIN, and the smaller of the two will be the LEVITATE time step for the next calculational cycle. It is noted that the LEVITATE time step is not allowed to exceed the time remaining until the end of the heat-transfer time step. If this happens, the LEVITATE time step will be cut back, so that the end of the next LEVITATE time step will coincide with the end of the heat-transfer time step. The next task of LEVDRV is to calculate the data for the fuel, steel and sodium reactivities. These calculations are described in more detail in the next section on LEVITATE interaction with the FEEDBK routine.

If the end of the LEVITATE time step coincides with the end of a heat-transfer time step the PLHTR routine is called. This routine calculates the new temperatures in the solid fuel pin and in the cladding outside the interaction region at the end of the current heat transfer time step. Then a new heat transfer time step is calculated.

If it is time to produce output, LEVDRV will print the output described in Section 16.9. Then if the end of the LEVITATE time step does not coincide with the end of the primary loop time step, a new computational cycle begins. Otherwise LEVDRV returns control to TSTHRM.

16.1.3.3 LEVITATE Interfaces

During the time period when LEVITATE performs calculations in a given channel, it interacts with the PRIMAR hydrodynamic model, the PLHTR heat transfer model, and the FEEDBK reactivity model (Figure 16.1.4).

16.1.3.3.1 LEVITATE Interface with the PRIMAR Hydrodynamic Model

PRIMAR calculates the hydrodynamic behavior of the sodium loop outside the reactor vessel. Two options are now available, PRIMAR-1 and PRIMAR-4. PRIMAR-1, which can be selected by setting the input IPRION not equal to 4, calculates the inlet and outlet sodium pressures, using the pump flow decay curve, without accounting for the pressure events occurring in the LEVITATE region. Thus, no feedback is returned from LEVITATE to PRIMAR-1. LEVITATE uses the pressures calculated by PRIMAR-1, together with the pressures in the voided region to model the motion of the liquid sodium slugs which bound the voided region. The quantities PREX and PRIN, which in LEVITATE designate the upper and lower plenum pressures, respectively, are set in LEVDRV, using the pressures PX and PIN received from PRIMAR-1. PRIMAR-4 is a more sophisticated model, which accounts for the events calculated by LEVITATE. LEVITATE receives the inlet and outlet pressures from PRIMAR-4, as well as the time derivatives of these pressures. This information is used to determine the behavior of the sodium slugs. It also integrates the mass flow rates for the two slugs over the PRIMAR-4 time step and returns this information to PRIMAR-4. These calculations are performed in the LEVITATE routine LEMOCO, after solving the momentum conservation equations. The PRIMAR-4 model can be selected by setting the input variable IPRION equal to 4.

16.1.3.3.2 LEVITATE Interface with the PLHTR Heat-Transfer Module

PLHTR calculates the temperature transients in the fuel pin. As such, it interfaces with LEVITATE at the pin cavity surface and at the outer fuel-pin surface.

A heat flux, based on the temperature difference between the temperature of the pin inner node and the cavity temperature and an appropriate heat-transfer coefficient is calculated in the routine LElPIN. This flux is then used in LEVITATE as a boundary condition to calculate the fuel temperature in the pin cavity. The same flux is integrated over the heat-transfer time step and the resulting energy HFCAWA is then made available to the PLHTR module, for use as the boundary condition on the cavity side in the transient pin temperature calculation.

A similar heat flux, based on the temperature difference between the fuel-pin outer node and the inner node of the cladding is calculated in the LEMISC routine. The heat-transfer coefficient used is generally based on the gap conductance. A more complex situation exists when the cladding has been completely ablated at a certain axial location. In this case, the fuel is in direct contact with the materials in the channel, and the heat-transfer coefficient and temperature difference used are dependent on the local configuration. The heat flux calculated is then used as a boundary condition in LEVITATE for the calculation of the transient cladding temperature, or, when the cladding has been ablated, for the calculation of the temperatures in the channel. This heat flux is also integrated over the heat-transfer time step, and the resulting energy, HFPICL is then returned to the PLHTR module, where it is used as the outer-pin boundary condition in the transient pin temperature calculation.

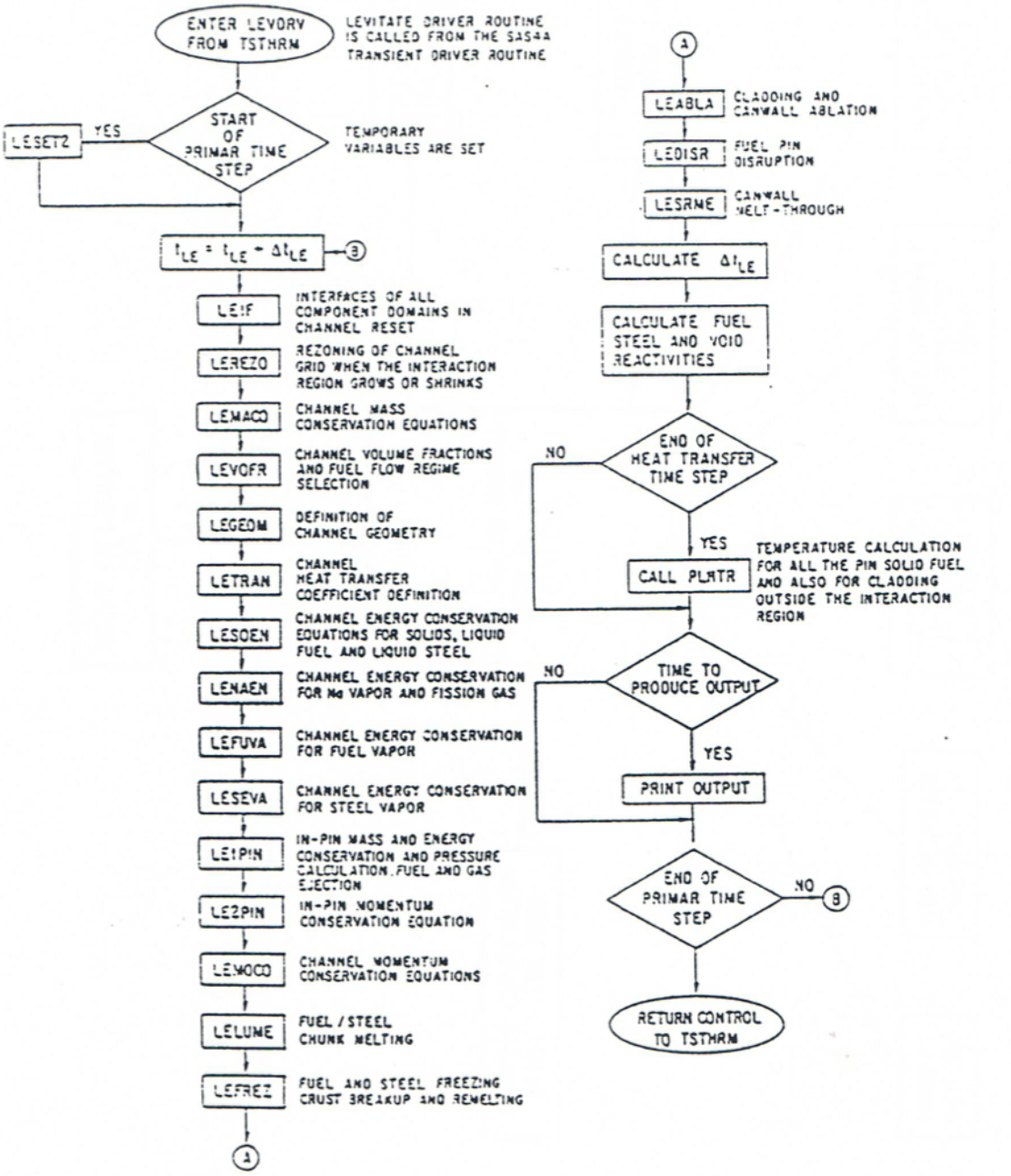


Figure 16.1.5: LEVITATE Driver Flowchart

16.1.3.3.3 LEVITATE Interface with the FEEDBK Reactivity Model

FEEDBK calculates the net reactivity changes for the whole reactor during a time step and transfers this information to the neutronic model which calculates the changes in the reactor power. LEVITATE calculates the sodium, fuel and steel axial mass distributions for the SAS4A channel under its control at the end of each time step. In other channels, these masses can be updated by other modules, e.g., PLUTO2, which have control at the given time. The transient axial mass distributions are used in subroutine FEEDBK to calculate the coolant void, cladding motion, and fuel motion reactivity feedbacks. When all channels have been calculated, the material relocation reactivities are used in FEEDBK, together with the Doppler reactivity, to determine the total reactivity of the core. Using this information, the neutronic model determines the new power level which is used by LEVITATE in the following time step. The power level at the end of each LEVITATE time step is determined in LEVDRV, using an exponential fit of the power-time history supplied by the neutronic model. This fit is based on the power level at the beginning of the previous main time step, the power level at the beginning of the current main time step and the precalculated power level at the end of the current main time step. By using this calculated power level and the axial input power distribution, the specific power for each axial cell is calculated.

16.1.3.4 LEVITATE Termination

At the present time, LEVITATE is the last phenomenological module called by SAS4A. Thus the decision to terminate a SAS4A run will generally be made in LEVITATE. This decision is made in the routine LEABLA, where the number of completely molten hexcan wall cells are counted. If this number $N_{sr,melt}$ is greater or equal to an input specified number, NSLEEX, LEVITATE will print an explanatory message and will terminate the calculations. The reason behind this is that once the hexcan walls have been totally molten at a certain axial location it is likely that inter-channel lateral material exchanges will be initiated, i.e., a transition-phase domain is being entered that is not modeled in SAS4A.

Other possibilities to terminate the SAS4A calculations, which are not necessarily specific to LEVITATE are: (a) exceeding the maximum number of main time steps in the transient calculation MAXSTP; (b) exceeding the maximum problem time TIMAX; and (c) decreasing the fuel motion reactivity below the input value NFUELD.

16.1.4 Improvements and New Models Relevant to the LEVITATE Module in SAS4A Version 1.1

This section describes the differences between the LEVITATE module present in the SAS4A Release 1.1 version and that used in the SAS4A Release 1.0.

The most important enhancement to the LEVITATE Release 1.1 is the availability of the chunk model, which describes the formation and relocation of solid fuel and/or steel chunks. Although the chunk model was described in the documentation to Release 1.0, the actual code did not incorporate this capability which was still being tested at the time of release. The chunk modeling capability has been fully integrated in the Release 1.1 LEVITATE module. However, the user should be aware that this model has not yet been validated by experiment analysis and its results should be used cautiously at this time.

The chunk model can be switched on by setting the input variable **ICHUNK**=1. If ICHUNK is set to zero, the chunk model is completely disabled and the code will work in the same manner as in Release 1.0, i.e., will homogenize the moving solid fuel with the liquid fuel in the channel. The new input variables relevant for the chunk model are **ICHUNK**, **ILUBLK**, **ASRALU**, **RALUDI**, and **RALUFZ**. These variables are described in Table 16.8.1.

Another feature added to the current LEVITATE version is the presence of a partial annular steel flow regime. This flow regime is described in Section 16.4.1.4. The partial annual steel flow regime is used only when the chunk model is used, i.e., ICHUNK=1. If ICHUNK=0 the full annular flow regime is still used. It should be

noted here that the annular steel flow regime becomes more important when the chunk model is used, because the frozen fuel can separate from the molten fuel-steel field, leading to the presence of more cells characterized by the steel annular flow regime.

A mechanistic model describing the ejection of the molten fuel from the pin cavity into the coolant channel has also been added to the LEVITATE module. This model is optional and can be used by setting the input variable INRAEJ=1. The velocity of the ejected fuel is calculated using a radial momentum conservation equation. The radial momentum can also be convected axially, in axial cells where the cladding failure has occurred.

A complete calculation of the sodium and structure temperatures in the sodium slug region has also been added and the corresponding temperature map is now part of the LEVITATE regular output.

16.2 In-pin Hydrodynamic Model

16.2.1 Physical Models

The in-pin hydrodynamic model describes the motion of the molten fuel and fission gas mixture in the cavity formed inside the fuel pins during a loss of flow accident. As the accident proceeds, the size of the cavity increases, both radially and axially (Figure 16.1.2). Newly molten fuel and fission gas are added to the moving components in the cavity. Some of the fission gas is dissolved in the molten fuel, in the form of small bubbles constrained by surface tension. The effect of this fission gas is controlled by the input variable PRSFTN. If PRSFTN is less than 10^7, the volume of the dissolved gas is assumed to be negligible, and thus it does not contribute immediately to the cavity pressure. If PRSFTN $> 10^7$, the volume occupied by the dissolved gas is taken into account in the cavity pressure calculation, as described in Section 14.2.6. The remainder of the fission gas is in the form of free gas, residing in bubbles that are too large to be constrained by surface tension. This gas contributes immediately to the pressurization of the cavity. Because of the continuous coalescence of the small bubbles, leading to the formation of new larger bubbles, the originally dissolved gas is continuously released from the molten fuel and is added to the free fission gas. The continuous heating of the molten fuel and fission gas leads to the pressurization of the cavity and eventually to the cladding failure. The fuel and fission gas in the vicinity of the failure location are ejected into the coolant channel leading to a local depressurization of the cavity. This depressurization causes the fuel motion inside the pin toward the failure location.

The in-pin fuel motion is treated as a one-dimensional, compressible flow with a variable flow cross section. The fuel and fission gas are assumed to form a homogeneous mixture in thermal equilibrium. However, if the local fuel volume fraction is less than an input value FNFUAN, the pressure gradient is assumed to act only on the fuel cross section. This is an attempt to roughly account for the annular fuel flow regime which may exist for large void fractions. Fuel vapor pressures are also included in the in-pin hydrodynamic model. The fuel vapor pressure is based on the average fuel temperature in any axial cell, with the assumption that no significant radial temperature profiles can be maintained after the onset of fuel motion.

As the cladding heats up and the in-pin pressures increase, the cladding rip can propagate from the original location, allowing the ejection of fuel and fission gas from the cavity into the channel at new axial locations. The fuel pin can also be totally disrupted at certain axial locations where the cladding becomes very weak and the solid pin is largely molten. The pin disruption leads to the formation of upper and lower pin stubs with cavities that can eject fuel axially into the disrupted region.

16.2.2 Method of Solution and General Numerical Considerations

16.2.2.1 Variables and Mesh Grid Used in Calculations

The independent variables used in the in-pin model are the axial coordinate z and the time t. Only one spatial coordinate is necessary, as LEVITATE models the pin cavity in a one-dimensional geometry. The dependent variables calculated by the in-pin hydrodynamic model for each component are the generalized density ρ', the enthalpy h (or temperature T), and the velocity u. The generalized densities have been introduced in Chapter 14 and, for the component j in the cavity, are defined as follows:

$$\rho'_{\text{j,ca,k}} = \rho_{\text{j,ca,k}} \cdot \frac{A_{\text{j,ca,k}}}{\text{AXMX}} = \rho_{\text{j,ca,k}} \cdot \theta_{\text{j,ca,k}} \tag{16.2-1}$$

where

$\rho_{\text{j,ca,k}}$ is the physical density of component j at the axial location k in the pin cavity

$A_{\text{j,ca,k}}$ is the cross sectional area occupied by component j at location k in the pin cavity. This area refers to all the pins in the subassembly.

AXMX is the reference input area

$\theta_{\text{j,ca,k}}$ is the generalized volume (or area) fraction of component j at location k in the pin cavity

The mass, energy and momentum partial differential equations are solved using the Eulerian finite difference semi-explicit formulation, as explained in Section 16.2.2.2. A staggered mesh grid is used to obtain the numerical formulation, with the densities and enthalpies defined at the center of each cell while the velocities are defined at the boundaries. Variable flow areas are treated in the in-pin model. However, the cavity geometry is not as irregular as the coolant channel geometry, and the conventional single velocities were used at the boundaries, as opposed to the dual velocities used in the coolant channel model (see Section 16.4.2.1). The components treated in the cavity model are the molten fuel, the dissolved fission gas and the free fission gas. They are assumed to form a homogeneous mixture and move with the same velocity at all axial locations. A full donor cell formulation was used in the finite difference formulation in order to improve the stability characteristics of the solution.

16.2.2.2 Description of the Method of Solution and Logic Flow

The in-pin hydrodynamic model is solved in the routines LE1PIN and LE2PIN. First, the cavity enlargement is calculated in LE1PIN, using the solid fuel temperatures provided by PLHTR. If melting occurs, the quantities of molten fuel and fission gas to be added to the moving components are determined. Then the mass conservation equation is solved for all axial cells, accounting for the changes in fuel and free fission-gas mass due to convection and melt-in. LE1PIN then solves the energy conservation equations for the fuel and fission-gas mixture. It is assumed that the fuel and fission gas remain at the same temperature in all the axial cells. Using the new masses and temperatures, the new pressures are then calculated. It is noteworthy that at this point the pressures have also been updated in the channel (see Section 16.4.2.2) so that we can use a consistent set of pressures for the ejection calculation, which is the next step in the LE1PIN routine. The ejection calculation leads to changes in the mass and pressure in all nodes that are ejecting fuel and fission gas into the coolant channel. The ejection can take place radially, via the cladding rip or axially, via the open ends of the fuel-pin stubs when the fuel-pin disruption has already occurred. Both modes of ejection can be present simultaneously. The LE1PIN routine also examines the possibility of pin disruption. If an axial cell is found which has to be disrupted, a flag is set (IDISR(I)=9), but no other changes are preformed. The actual pin disruption is performed in the routine LEDISR, as described in Section 16.5.

The routine LE2PIN then solves the momentum conservation equation for all cells, obtaining the new velocities at the end of the time step. These velocities are obtained by using the new pressures calculated in LE1PIN, and in this sense, the method of solution is mixed, explicit-implicit, rather than purely explicit. The routine

LE2PIN also solves the mass conservation equations for the dissolved gas and calculates the maximum time step acceptable for the in-pin hydrodynamic model in the next computational cycle.

16.2.3 Finite Difference Forms and Solution Technique, Special Situations

The equations describing the in-pin hydrodynamic and thermal process are solved in the LE1PIN and LE2PIN routines. The partial differential equations, as well as the finite difference formulation, are generally the same in LEVITATE and PLUTO2. The reader is referred to the PLUTO2 chapter (Section 14.2) for a detailed description of the equations. Only the features of the in-pin model that are specific to LEVITATE will be discussed here.

The main feature of LEVITATE is the fuel-pin disruption mode. The decision for disruption of a certain axial pin cell is made in the routine LE1PIN. An undisrupted cell will be disrupted if the molten fuel cavity covers a large fraction of the original pin cross section.

$$\frac{R_{\text{ca,k}}}{R_{\text{pin,os,k}}} > \text{FNDISR} \tag{16.2-2}$$

and the cladding is molten or close to melting, i.e.,

$$T_{\text{cl,os,i}} > T_{\text{se,so}} \tag{16.2-3}$$

and:

$$T_{\text{cl,in,i}} > T_{\text{se,so}} - 50 \quad \text{if } \Delta R_{\text{cl,k}} > 0.5 \cdot \Delta R^0_{\text{cl}} \tag{16.2-4}$$

or:

$$T_{\text{cl,in,i}} > T_{\text{se,so}} - 150 \text{if } 0 < \Delta R_{\text{cl,k}} < 0.5 \Delta R^0_{\text{cl}} \tag{16.2-5}$$

The disruption of one or more axial pin nodes leads to the formation of a disrupted region, extending from the cell IDISBT to IDISTP. Note that the axial position of the in-pin cells is denoted in Eq. (16.2-1) by the subscript k, while the subscript used in the channel is i. The correspondence between i and k is given below:

$$i = k + \text{IDIFF} \tag{16.2-6}$$

that is, the in-pin cell k will have the same axial location as the channel cell $k + \text{IDIFF}$.

The motion of the material present in this region is calculated by the coolant channel hydrodynamic model. In the disrupted region, the coolant channel covers the entire cross sectional area of the subassembly. Only one disrupted region is allowed in LEVITATE. Thus, if two or more disjoint disrupted regions appear at any given time (e.g., cells 8 and 10 are disrupted, but not 9), the undisrupted nodes between the regions will also be forced to disrupt. In this case, a message is printed indicating that one or more nodes have been disrupted due to the disruption of neighboring nodes and the formation of large fuel/steel chunks is likely to occur. Only the decision about disruption is made LE1PIN. The nodes to be disrupted are flagged (IDISR $(I) = 9$), but the disruption process will only be modeled later in the cycle, in the routine LEDISR. This process is described in Section 16.3.

After the occurrence of the fuel-pin disruption, the model describing the hydrodynamics of the molten fuel/fission gas in the pin cavity continues to operate. However, instead of modeling a continuous channel, it now describes the hydrodynamic behavior of two disjoint channels, i.e., the cavities remaining in the upper and lower undisrupted pin stubs. These cavities communicate directly with the channel via the open ends of the pin stubs. The channel pressure in front of these open ends is used as the boundary condition for the in-pin hydrodynamic model.

16.3 Fuel Ejection from the Pins and Fuel-pin Disruption

The hydrodynamic in-pin model is connected to the channel model via the mechanisms of molten fuel ejection into the channel and fuel-pin disruption. The models describing these processes are described below.

16.3.1 Fuel Ejection via a Cladding Rupture

The molten fuel/fission-gas mixture can be ejected from the pin cavity into the channel in any axial cell where a cladding rupture has already occurred. The original cladding failure location is determined in the routines FAILUR and FUINIT, just before LEVITATE is initiated. Once control passes to LEVITATE, additional nodes can fail, leading to the increase of the original failure and, occasionally, the initiation of new failures. Thus, two or more disjoint failures can be present in a channel under certain circumstances. The fuel-pin rip enlargement is performed in the routine LEIF.

The stress in the cladding in the axial cell is calculated as follows:

$$\sigma_{\text{cl,i}} = \frac{P_{\text{ca,k}} \cdot R_{\text{ca,k}} - P_{\text{cl,i}} \cdot R_{\text{cl,os,i}}}{\Delta R_{\text{cl,i}}} \tag{16.3-1}$$

The ultimate tensile strength for the cladding is calculated using the function $\sigma_{\text{U}}(T)$ for each radial cladding node in the axial cell i:

$$\sigma^{U}_{\text{cl,i}} = \frac{\sigma^{U}(T_{\text{cl,os,i}}) \cdot (R_{\text{cl,os,i}} - R_{\text{cl,in,i}}) + \sigma^{U}(T_{\text{cl,in,i}}) \cdot (R_{\text{cl,in,i}} - R_{\text{cl,is,i}})}{(R_{\text{cl,os,i}} - R_{\text{cl,is,i}})} \tag{16.3-2}$$

The rupture will be extended to cell i if:

$$\sigma_{\text{cl,i}} > \sigma^{U}_{\text{cl,i}} \tag{16.3-3}$$

during any time step. All the nodes in the LEVITATE region are scanned every cycle. Once the rupture has occurred in one axial node, that node will continuously be checked for fuel ejection in each cycle.

The ejection of the molten fuel from the pin cavity into the channel is performed in the LE1PIN routine. Only the nodes where the cladding has been ruptured can eject fuel. Ejection will occur in these nodes if:

$$P_{\text{ca,K}} > P_{\text{ch,i}} \tag{16.3-4}$$

and

$$\alpha_{\text{fu,ca,k}} = \frac{\theta_{\text{fu,ca,k}}}{\theta_{\text{ca,k}}} > 0.1 \tag{16.3-5}$$

An additional constraint, designed to avoid numerical problems related to a fully incompressible channel configuration, is:

$$\frac{(\theta_{\text{ch,i}} - \theta_{\text{ch,op,i}}) + \theta_{\text{fu,i}} + \theta_{\text{se,i}} + \theta_{\text{fu,i}}}{\theta_{\text{ch,i}}} \leq 0.95 \tag{16.3-6}$$

i.e., the volume fraction of the incompressible components (crust, fuel, steel, and chunks) should not be more than 95% of the original channel volume. If all the conditions 16.3-4 through 16.3-6 are satisfied, the injection calculation begins by calculating the in-pin pressure after injection. An estimate of the amounts of fuel that can be ejected during a typical LEVITATE time step indicated that at the time of failure, the limiting factor in the ejection process is the inertia for the molten fuel. For the time steps usually used in LEVITATE (approximately 2×10^{-5} s) and the typical conditions for a loss-of-flow situation, where the coolant channel is voided, it was found that the pressure in the pin cavity will decrease initially by about 0.5% of the pressure difference between the

cavity and the channel. However, the fuel in the cavity can be accelerated laterally quite rapidly, and the inertial constraint becomes insignificant within a few milliseconds of the failure time. Afterwards, the ejection process is rapid enough to equilibrate the pressures in the cavity and channel within one or two typical LEVITATE time steps.

$$P_{\text{ca,k}}^{n+1} = P_{\text{ca,k}}^{n} - (P_{\text{ca,k}} - P_{\text{ch,i}}) \cdot \text{CIPINJ} \cdot \text{DTPLU} \tag{16.3-7}$$

The input constant CIPINJ has the recommended value 2.5×10^4. For this value of CIPINJ, the product CIPINJ $\times$ DTPLU has the value 0.5 and the pressure difference between the cavity and the coolant channel will be equilibrated very rapidly. This model does not account for the inertial effects mentioned above, which are present for a short time after the failure has occurred. A more mechanistic model accounting for these effects is currently being developed. With the assumption that the fuel, fission gas and fuel vapor in the axial cell k are homogeneously mixed, we can now calculate the amount of fuel which has to be ejected from the cavity in order to establish the pressure $P_{\text{ca,k}}^{n+1}$:

$$\begin{aligned} \Delta\rho'_{\text{fu,k}} = \{&[\theta_{\text{fi,ca,k}} - \theta_{\text{fu,ca,k}} \cdot \text{CMFU} \cdot (P_{\text{ca,k}} - P_{\text{fv,ca,k}})] \cdot P_{\text{ca,k}} \\ &+ \left(R_{\text{fi}} \cdot \rho'_{\text{fi,ca}} + R_{\text{fv}} \cdot \rho'_{\text{fv,ca,k}}\right) \cdot T_{\text{fu,ca,k}}\} \cdot \rho_{\text{fu,ca,k}} \\ &\div (P_{\text{ca,k}} + f_{\text{fi,fu,k}} \cdot R_{\text{fi}} \cdot T_{\text{fu,ca,k}} \cdot \rho_{\text{fu,ca,k}} \\ &+ f_{\text{fv,fu,k}} \cdot R_{\text{fv}} \cdot T_{\text{fu,ca,k}} \cdot \rho_{\text{fu,ca,k}}) \end{aligned} \tag{16.3-8}$$

where the input variable CMFU represents the compressibility of the molten fuel and

$$f_{\text{fi,fu,k}} = \frac{\rho'_{\text{fi,ca,k}}}{\rho'_{\text{fu,ca,k}}} \tag{16.3-9}$$

$$f_{\text{fi,fu,k}} = \frac{\rho'_{\text{fv,ca,k}}}{\rho'_{\text{fu,ca,k}}} \tag{16.3-10}$$

In the derivation of Eq. (16.3-8), we used the assumption that, for injection purposes, the fuel vapor behaves as a gas. Further fuel vaporization can take place in the next cycle. It should also be noted that $\Delta\rho'_{\text{fu,k}}$ represents the change in the generalized density of the in-pin fuel. This change refers to the fuel in all the pins in the subassembly, as explained in the corresponding PLUTO2 section.

The amount of fission gas and fuel vapor ejected is then calculated as

$$\Delta\rho'_{\text{fi,k}} = f_{\text{fi,fu,k}} \cdot \Delta\rho'_{\text{fu,ca,k}} \tag{16.3-11}$$

$$\Delta\rho'_{\text{fv,k}} = f_{\text{fv,fu,k}} \cdot \Delta\rho'_{\text{fu,ca,k}} \tag{16.3-12}$$

The fuel vapor mass ejected is then added to the mass of the liquid fuel ejected. It is noteworthy that the fuel vapor pressure is used only for the momentum calculation in the pin and for the fuel ejection. No fuel vapor conservation equations are solved in the pin cavity.

The new channel partial pressures are then recalculated:

$$P_{\text{fv,i}} = \frac{R_{\text{fv}} \cdot \rho'_{\text{fv,i}} \cdot T_{\text{fv,i}}}{\theta_{\text{vg,i}}^{n+1}} \tag{16.3-13}$$

$$P_{\text{sv,i}} = \frac{R_{\text{sv}} \cdot \rho'_{\text{sv,i}} \cdot T_{\text{sv,i}}}{\theta_{\text{vg,i}}^{n+1}} \tag{16.3-14}$$

If $\frac{\theta_{\text{Na,lq,i}}}{\theta_{\text{ch,op,i}}} \leq 0.3$

$$P_{\text{fi,i}} = \frac{R_{\text{fi}} \cdot \left(\rho'_{\text{fi,i}} + \Delta\rho'_{\text{fi,k}}\right) \cdot T_{\text{fi,i}}}{\theta_{\text{vg,i}}^{n+1}} \tag{16.3-15}$$

If $\frac{\theta_{\text{Na,lq,i}}}{\theta_{\text{ch,op,i}}} > 0.3$, then

$$P_{\text{fi,i}} = \frac{-\theta_{\text{vg,i}}^{n+1} + \sqrt{\left(\theta_{\text{vg,i}}^{n+1}\right)^2 + 4\theta_{\text{Na,}lq\text{,i}} \cdot \text{CMNL} \cdot \left(\rho'_{\text{fi,i}} + \Delta\rho'_{\text{fi,ca,k}}\right) \cdot R_{\text{fi}} \cdot T_{\text{fi,i}}}}{2\theta_{\text{Na,}lq\text{,i}} \cdot \text{CMNL}} \tag{16.3-16}$$

If $\theta_{\text{Na,}lq} = 0$, i.e., if the sodium present is in the form of single-phase vapor:

$$P_{\text{Na,i}} = \frac{R_{\text{Na}} \cdot \rho'_{\text{Na,i}} \cdot T_{\text{Na,i}}}{\theta_{\text{vg,i}}^{n+1}} \tag{16.3-17}$$

If $0 < \theta_{\text{Na,}lq}$, i.e., two-phase sodium is present:

$$P_{\text{Na,i}} = P_{\text{Na}}\left(T_{\text{Na,i}}\right) \tag{16.3-18}$$

where CMNL is the compressibility coefficient of the liquid sodium and $\theta_{\text{vg,i}}^{n+1}$ is:

$$\theta_{\text{vg,i}}^{n+1} = \theta_{\text{vg,i}} - \frac{\Delta\rho'_{\text{fu,ca,k}}}{\rho_{\text{fu,k}}} \tag{16.3-19}$$

The new tentative channel pressure is:

$$P_{\text{ch,i}}^{n+1} = P_{\text{Na,i}} + P_{\text{fi,i}} + P_{\text{sv,i}} + P_{\text{fv,i}} \tag{16.3-20}$$

If this pressure is below the pressure in the cavity after injection, i.e.,

$$P_{\text{ch,i}}^{n+1} < P_{\text{ca,k}}^{n+1} \tag{16.3-21}$$

then no further iterations are necessary and $P_{\text{ch,i}}^{n+1}$ remains the new pressure in the channel. If Eq. (16.3-21) is not satisfied, too much material was ejected from the cavity into the channel. A new pressure in the pin cavity is selected:

$$P_{\text{ca,k}}^{n+1} = \left(P_{\text{ca,k}}^{n} + P_{\text{ca,k}}^{n+1}\right) \cdot 0.5 \tag{16.3-22}$$

and the procedure outlined in Eq. (16.3-7) through Eq. (16.3-20) is repeated until the condition in 16.3-18 is satisfied.

It should be noted that in Eq. (16.3-15), the fission-gas pressure is calculated by assuming that the fission gas injected has the same temperature as the fission gas in the channel. This is necessary because the fission gas and sodium have to be at the same temperature, which has already been calculated in the routine LENAEN. The additional energy carried by the injected fission gas, which in fact has the temperature $T_{\text{fu,ca,k}}$ will be added to the fission gas/sodium mixture in the next cycle, when solving the energy equation for these components. Finally, the fuel and fission-gas densities in the pin and channel are updated.

$$\rho'_{\text{fu,ca,k}} = \rho'_{\text{fu,ca,k}} - \Delta\rho'_{\text{fu,ca,k}} \tag{16.3-23}$$

$$\rho'_{\text{fi,ca,k}} = \rho'_{\text{fi,ca,k}} - \Delta\rho'_{\text{fi,ca,k}} \tag{16.3-24}$$

$$\rho'_{\text{fu,i}} = \rho'_{\text{fu,i}} + \Delta\rho'_{\text{fu,ca,k}} \tag{16.3-25}$$

$$\rho'_{\text{fu,i}} = \rho'_{\text{fi,i}} + \Delta\rho'_{\text{fi,ca,k}} \tag{16.3-26}$$

In the Release 1.1 version of LEVITATE a mechanistic calculation of the in-pin fuel ejection has been added. This model can be used by setting the input variable **INRAEJ**=1. If INRAEJ=0 the ejection of the fuel into the channel is calculated as described above.

The mechanistic ejection model calculates the radial velocity of the in-pin molten fuel/gas mixture, $u_{\text{ca-ch,k}}$, at each axial location where the cladding has been ruptured. This velocity is recalculated each time step. The amounts of fuel and fission gas ejected each time step are calculated as follows:

$$\Delta\rho'_{\text{fu,k}} = \rho'_{\text{fu,k}} \cdot \frac{V_{\text{fuel ejected}}}{V_{\text{fuel cavity}}} = \rho'_{\text{fu,k}} \cdot \frac{u_{\text{fu,ca-ch,k}} \cdot \Delta t}{\pi \cdot R_{\text{ca,k}}} \tag{16.3-27}$$

$$\Delta\rho'_{\text{fi,k}} = \rho'_{\text{fi,k}} \cdot \frac{u_{\text{fu,ca-ch,k}} \cdot \Delta t}{\pi \cdot R_{\text{ca,k}}} \tag{16.3-28}$$

$$\Delta\rho'_{\text{fv,k}} = \rho'_{\text{fv,k}} \cdot \frac{u_{\text{fu,ca-ch,k}} \cdot \Delta t}{\pi \cdot R_{\text{ca,k}}} \tag{16.3-29}$$

The pressure in the cavity is then calculated solving equation 16.3-8 for $P_{\text{ca,k}}$ and the pressure in the coolant channel is calculated using Eq. (16.3-13) through Eq. (16.3-20). If

$$P^{n+1}_{\text{ch,i}} < P^{n+1}_{\text{ca,k}} \tag{16.3-30}$$

then no further iterations are necessary. If 16.3-23-4 is not satisfied then too much material was ejected from the cavity into the channel. The radial velocity is reduced:

$$u^{n+1}_{\text{fu,ca-ch,k}} = u^{n+1}_{\text{fu,ca-ch,k}} \cdot 0.5 \tag{16.3-31}$$

and the procedure for pressure calculation is repeated.

The key element in the calculation described above is the radial fuel velocity $u_{\text{fu,ca-ch,k}}$, which is calculated as described below.

The radial velocity calculation first updates all radial velocities considering only the radial acceleration $a_{\text{fu,ca-ch,k}}$, and ignoring temporarily the axial transport of radial momentum:

$$u^{n+1}_{\text{fu,ca-ch,k}} = u_{\text{fu,ca-ch,k}} + a_{\text{fu,ca-ch,k}} \cdot \Delta t \tag{16.3-32}$$

where $a_{\text{fu,ca-ch,k}}$ is the radial acceleration of the molten fuel in the cavity in the axial cell k. This acceleration is calculated using the simplified geometry illustrated in Figure 16.3.1. The pressure difference between the pin cavity and the coolant channel is assumed to act on the fuel contained in the shaded volume. The rip size is considered to be of the same order of magnitude as the cavity radius. The acceleration is calculated as follows:

$$\begin{aligned} a_{\text{fu,ca-ch,k}} = \frac{\sum \text{Forces}}{\text{Mass}} = \frac{R_{\text{ca,k}} \cdot z_{\text{k}}}{0.9 \cdot R^{z}_{\text{ca,k}} \cdot z_{\text{k}} \cdot \rho_{\text{fu,k}}} \{ & P_{\text{ca-ch,k}} - 0.5 \cdot \rho_{\text{fu,k}} \cdot u_{\text{fu,ca-ch,k}} \\ & \cdot |u_{\text{fu,ca-ch,k}}| \cdot C_{\text{ORIFICE}} - C_{\text{SHEAR}} \cdot \eta_{\text{fu,k}} \cdot \frac{u_{\text{fu,ca-ch,k}}}{R_{\text{ca,k}}} \Big\} \end{aligned} \tag{16.3-33}$$

where: $\Delta P_{\text{ca-ch,k}} = P_{\text{ca,k}} - P_{\text{ch,i}}$

C_{ORFICE} is an orifice coefficient used in the calculation of the flow pressure drop across the rip. This coefficient is currently zero in the code.

C_{SHEAR} is a coefficient associated with the shear forces exerted on the accelerating fuel. It can be determined from geometrical considerations. The current value is 6.88. $\eta_{\text{fu,k}}$ is the viscosity of the molten fuel in cell k

As the fuel from the radial control volume is ejected into the channel, it is replaced with new fuel from outside the control volume, assumed to carry no radial momentum. Thus, the radial velocity of the fuel has to be recalculated. into the channel, it is replaced with new fuel from outside the control volume, assumed to carry no radial momentum. Thus, the radial velocity of the fuel has to be recalculated.

$$\begin{aligned} u_{\text{fu,ca-ch,k}} &= u_{\text{fu,ca-ch,k}} \cdot \frac{V_{\text{control}} - V_{\text{ejected}}}{V_{\text{control}}} \\ &= u_{\text{fu,ca-ch,k}} \cdot \frac{0.9R_{\text{ca,k}} - u_{\text{fu,ca-ch,k}} \cdot \Delta t}{0.9R_{\text{ca,k}}} \end{aligned} \tag{16.3-34}$$

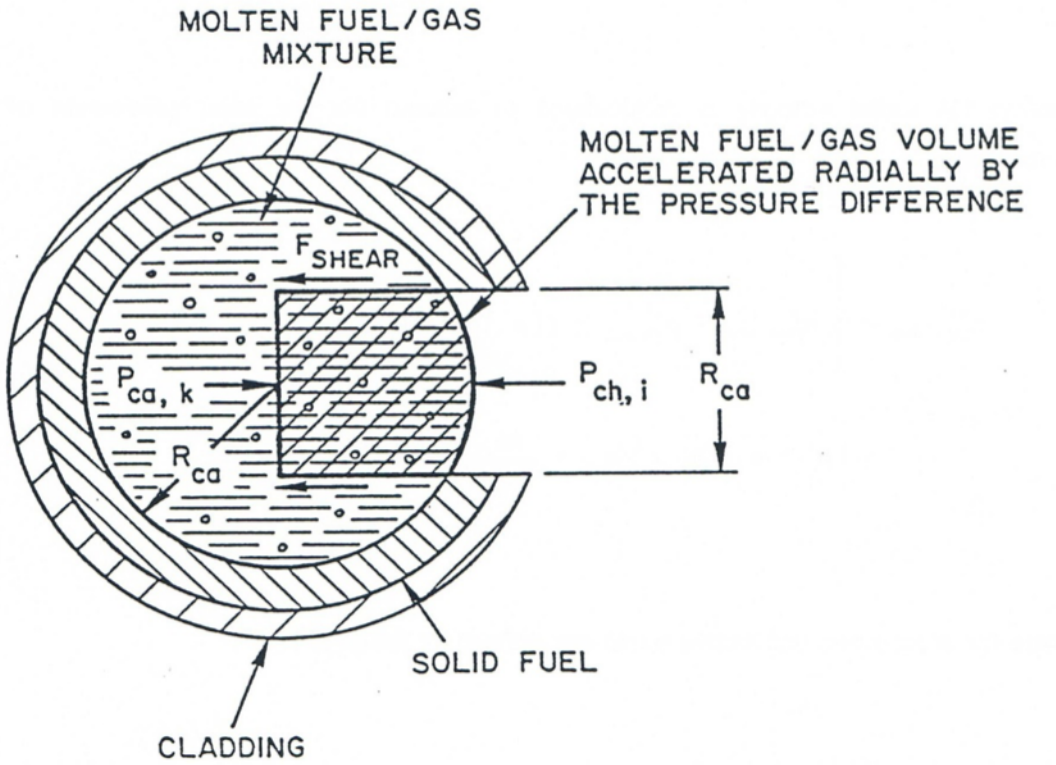


Figure 16.3.1: Geometry for the Radial Acceleration Calculation

Finally, the radial velocity is recalculated to account for the axial convection of radial momentum:

$$u^{n+1}_{\text{fu,ca-ch,k}} = \left\{ u^{n+1}_{\text{fu,ca-ch,k}} \cdot \rho'_{\text{fu,ca,k}} - \left([\rho' \cdot u \cdot u_{\text{ca-ch}}]_{\text{fu,k+1}} - [\rho' \cdot u \cdot u_{\text{ca-ch}}]_{\text{fu,k}} \right) \cdot \frac{t}{z_{\text{k}}} \right\} \cdot \frac{1}{\rho'_{\text{fu,ca,k}}} \tag{16.3-35}$$

where the momentum convective terms are defined as follows:

$$[\rho' \cdot u \cdot u_{\text{ca-ch}}]_{\text{fu,k}} = \begin{cases} \rho'_{\text{fu,ca,k-1}} \cdot u_{\text{fu,ca,k}} \cdot u_{\text{fu,ca-ch,k-1}} & \text{if } u_{\text{fu,ca,k}} \geq 0 \\ \rho'_{\text{fu,ca,k}} \cdot u_{\text{fu,ca,k}} \cdot u_{\text{fu,ca-ch,k}} & \text{if } u_{\text{fu,ca,k}} < 0 \end{cases} \tag{16.3-36}$$

The mechanistic ejection model has been compared with the pressure equilibration ejection model (INEAEJ $=$ 0) in several calculations. It appears that the pressure equilibration ejection model provides generally a good approximation to the results calculated by the mechanistic model. The larger differences appear during high power excursions and during the time immediately following the pin failure, when the inertial forces accounted for in the mechanistic model delay the early fuel ejection into the channel. It is expected that after validation in several experiment analyses the mechanistic ejection model will become the standard model to be used in SAS4A calculations.

16.3.2 Fuel-pin Disruption

When certain conditions are satisfied, the fuel pins can be totally disrupted at certain axial locations. The decision process which triggers the pin disruption is performed in the LE1PIN routine and has been outlined in Section 16.2. The nodes to be disrupted are flagged in LE1PIN by setting IFLAG $(I) = 9$. The disruption process itself is performed in the LEDISR routine and is described below.

The LEDISR routine first checks whether a node that is to be disrupted is within the physical region for fuel, fission gas, steel and fuel chunks, since all these components are likely to be created via the disruption process. If the node to be disrupted lies outside the fuel physical region, for example, the fuel region will be extended so that the molten fuel generated via the disruption process will be modeled appropriately. Then, the generalized densities and the temperatures of the components present in the channel are reset to account for the addition of mass due to the pin disruption:

$$\Delta\rho'_{\text{fu,i}} = \rho'_{\text{fu,ca,k}} \tag{16.3-37}$$

$$\Delta\rho'_{\text{fi,i}} = \rho'_{\text{fi,ca,k}} \tag{16.3-38}$$

$$\Delta\rho'_{\text{fl,i}} = \rho'_{\text{ffc,i}} + \frac{M_{\text{fu,i}} \cdot N_{\text{PIN}}}{\Delta z_{\text{i}} \cdot \text{AXMX}} \tag{16.3-39}$$

$$\Delta\rho'_{\text{sl,i}} = \rho'_{\text{sic,i}} + \frac{M_{\text{se,i}} \cdot N_{\text{PIN}}}{\Delta z_{\text{i}} \cdot \text{AXMX}} \tag{16.3-40}$$

where $M_{\text{fu,i}}$ and $M_{\text{se,i}}$ represent the mass of solid stationary fuel and steel, respectively, present in the axial node i, per pin. The temperatures are reset using an energy balance. For example, the new fuel enthalpy is given by:

$$h_{\text{fu,i}} = \frac{h_{\text{fu,i}} \cdot \rho'_{\text{fu,i}} + h_{\text{fu,ca,k}} \cdot \Delta\rho'_{\text{fu,i}}}{\rho'_{\text{fu,i}} + \Delta\rho'_{\text{fu,i}}} \tag{16.3-41}$$

The new fuel temperature is obtained from $h_{\text{fu,i}}$, using the function $T_{\text{fu}}(h)$. The velocities are reset using a momentum conservation equation. For example, the new fuel velocity at boundary i, when node i is disrupted,

is calculated as follows:

$$\begin{aligned} u''_{\text{fu,i}} = \big[& u_{\text{fu,i}} \cdot \left(\rho'_{\text{fu,i-1}} + \rho'_{sl,\text{i-1}}\right) \cdot 0.5 \cdot \Delta z_{\text{i-1}} \\ & + u''_{\text{fu,i}} \cdot \left(\rho'_{\text{fu,i-1}} + \rho'_{sl,\text{i-1}}\right) \cdot 0.5 \cdot \Delta z_{\text{i-1}} \\ & + u_{\text{fu,ca,k}} \cdot 0.5 \cdot \Delta\rho'_{\text{fu,i}} \cdot \Delta z_{\text{i}} \big] \\ \div \big[& \left(\rho'_{\text{fu,i-1}} + \rho'_{sl,\text{i-1}}\right) \cdot 0.5 \cdot \Delta z_{\text{i-1}} \\ & + \left(\rho'_{\text{fu,i}} + \rho'_{sl,\text{i}}\right) \cdot 0.5 \cdot \Delta z_{\text{i}} \\ & + \Delta\rho'_{\text{fu,i}} \cdot 0.5 \cdot \Delta z_{\text{i}} \big] \end{aligned} \tag{16.3-42}$$

Disruption of node i also affects the fuel velocity at the boundary $i+1$:

$$\begin{aligned} u''_{\text{fu,i+1}} = \big[& u'_{\text{fu,i+1}} \cdot \left(\rho'_{\text{fu,i}} + \rho'_{sl,\text{i}}\right) \cdot 0.5 \cdot \Delta z_{\text{i}} \\ & + u''_{\text{fu,i+1}} \left(\rho'_{\text{fu,i+1}} + \rho'_{sl,\text{i+1}}\right) \cdot 0.5 \cdot \Delta z_{\text{i+1}} \\ & + u_{\text{fu,ca,k}} \cdot 0.5 \cdot \Delta\rho'_{\text{fu,i}} \cdot \Delta z_{\text{i}} \big] \\ \div \big[& \left(\rho'_{\text{fu,i}} + \rho'_{sl,\text{i}}\right) \cdot 0.5 \cdot \Delta z_{\text{i}} \\ & + \left(\rho'_{\text{fu,i+1}} + \rho'_{sl,\text{i+1}}\right) \cdot 0.5 \cdot \Delta z_{\text{i+1}} \\ & + \Delta\rho'_{\text{fu,i}} \cdot 0.5 \cdot \Delta z_{\text{i}} \big] \end{aligned} \tag{16.3-43}$$

The velocities $u'_{\text{fu,i}}$ and $u''_{\text{fu,i}}$ represent the molten fuel velocities at the channel boundary i and are described in more detail in Section 16.4.2.1.

Finally, the densities are reset as follows:

$$\rho'_{\text{fu,i}} = \rho'_{\text{fu,i}} + \Delta\rho'_{\text{fu,i}} \tag{16.3-44}$$

$$\rho'_{\text{fi,i}} = \rho'_{\text{fi,i}} + \Delta\rho'_{\text{fi,i}} \tag{16.3-45}$$

$$\rho'_{fl,\text{i}} = \rho'_{fl,\text{i}} + \Delta\rho'_{fl,\text{i}} \tag{16.3-46}$$

$$\rho'_{sl,\text{i}} = \rho'_{sl,\text{i}} + \Delta\rho'_{sl,\text{i}} \tag{16.3-47}$$

The partial pressures are then reset to account for the new gas volume and gas densities in the disrupted node. A number of quantities related to the cladding are set to zero before the end of the routine. These quantities include the perimeter and thickness of cladding and the amount of fuel and steel in the cladding crust.

16.3.3 Fuel Ejection via the Open Pin Stubs

Once the fuel-pin disruption has occurred in a subassembly, the remaining pin stubs can eject fuel and fission gas axially into the disrupted region, in addition to the regular radial gas ejection. Fuel and fission gas can also reenter the fuel-pin stubs when the pressure in the disrupted region exceeds the pressure in the pin cavity. The ejection (or reentry) of the molten fuel via the open pin stubs is calculated by solving the momentum conservation equation for the cavity half-cells adjacent to the disrupted region. This equation is solved in the routine LE1PIN. It is derived in the same manner as described in the PLUTO2 chapter for all regular in-pin momentum cells, but using the cell geometry illustrated in Figure 16.3.2. The momentum cell for the bottom pin stub extends from the center of the cell IDISBT-1 to the end of the stub. The pressure difference used in the momentum equation is $P_{\text{IDISBT}} - P_{\text{ca,IDISBT-1}}$, where P_{IDISBT} is the pressure in the disrupted region the convective flux coming *from* the disrupted region is defined using the generalized density $\rho'_{\text{fu,IDISBT}}$, i.e., the fuel generalized density in the disrupted region.

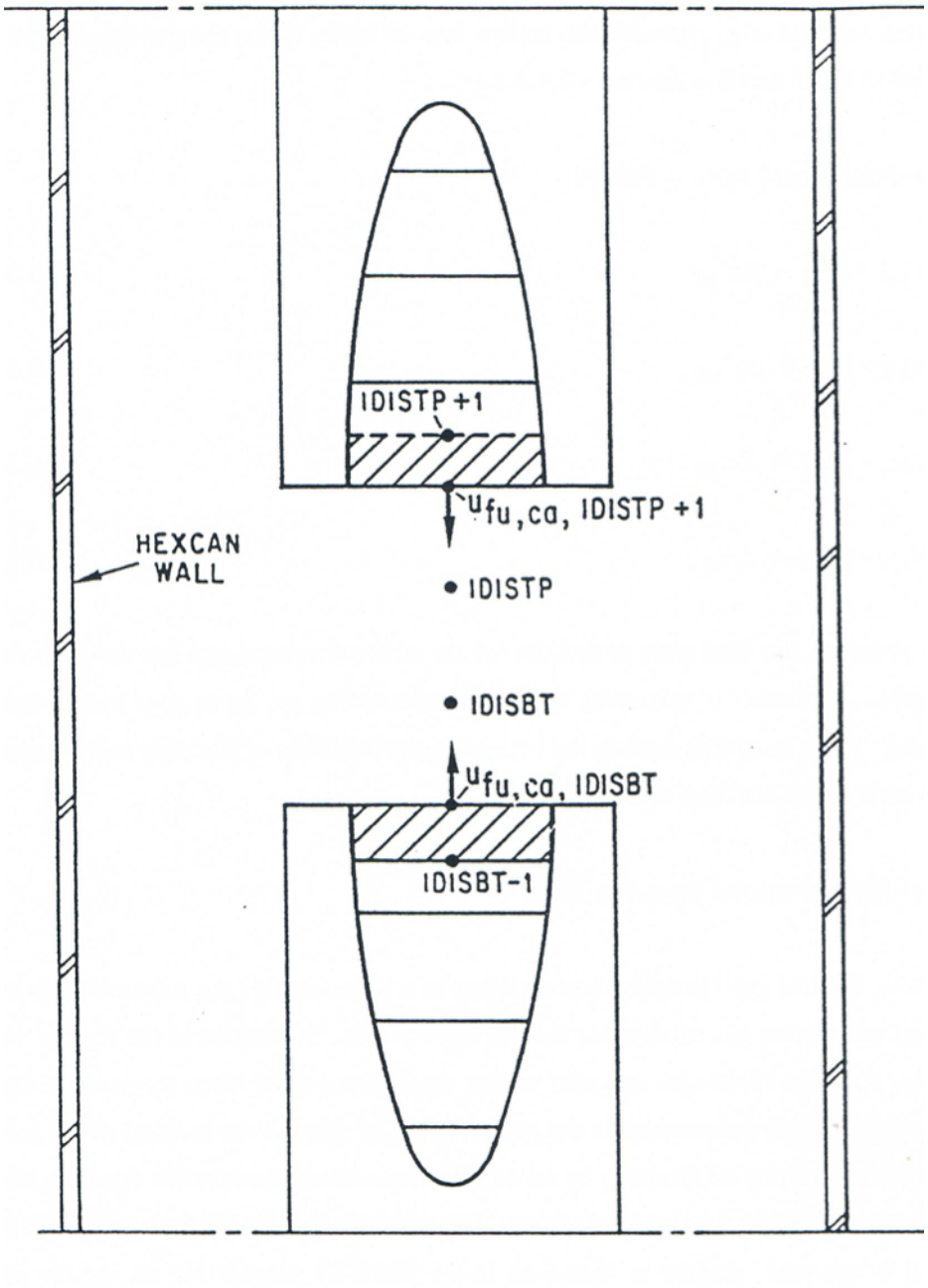


Figure 16.3.2: Geometry Used in the Derivation of the Momentum Equations for Ejection via the Pin Stubs

16.4 Coolant Channel Hydrodynamic Model

16.4.1 Physical Models

16.4.1.1 Geometry

The coolant channel described by LEVITATE is delimited by the hexcan wall and the outer surface of the fuel pins. Axially, this channel is limited by the upper and lower sodium plena. Calculations are performed in a 1-D geometry, with the variables being functions of the axial height z. As illustrated in Figure 16.1.1, the area of the coolant channel can vary widely due to the local disruption of the fuel pins, cladding and structure ablation, and fuel/steel blockage formation at various axial locations.

16.4.1.2 Hydrodynamic Models

The fluid dynamic models in LEVITATE calculate the motion of the materials in the channel. The materials that are tracked by LEVITATE are:

1. liquid fuel
2. liquid steel
3. sodium (liquid, two-phase mixture or superheated vapor)
4. fission gas
5. fuel vapor
6. steel vapor
7. solid fuel chunks
8. solid steel chunks

A separate mass and energy conservation equation is solved for each of these components. Momentum conservation equations are solved for three velocity fields, associated with the following component groups:

1. sodium, fission gas, fuel vapor, steel vapor
2. liquid fuel, liquid steel
3. solid fuel chunks, solid steel chunks

Phase transitions can occur, leading to mass, momentum and energy exchange among various moving components and/or among moving and stationary components. These exchanges are described in Section 16.4.1.5.

16.4.1.3 Structural Models

The structural models included in LEVITATE describe the thermodynamic behavior of the stationary components that act as boundaries for the coolant channel. The stationary components modeled in LEVITATE are:

1. steel cladding
2. steel hexcan wall
3. cladding frozen fuel crust (with possible steel inclusions)
4. hexcan wall frozen fuel crust (with possible steel inclusions)

One or more energy equations are solved for each of these components (e.g., two equations are needed for cladding, that has two radial nodes). Due to freezing/melting processes, mass and energy exchange can take place between these stationary components and the moving components in the channel, as explained in Section 16.4.1.5.

16.4.1.4 Fuel/Steel Flow Regimes

As indicated in Section 16.1, the interactions between the different components present in the channel, i.e., mass, energy and momentum transfer, are largely determined by the local configuration which, in turn, is determined by the flow regimes used. The fuel/steel flow regimes modeled in LEVITATE are presented conceptually in Figure 16.4.1. They are a bubbly fuel flow regime, a partial annular fuel flow regime, an annular steel flow regime, and a bubbly steel flow regime. The bubbly fuel channel flow is characterized by the presence of large amounts of liquid fuel, with droplets of molten steel, solid fuel and steel chunks, and bubbles of fission gas and sodium vapor imbedded. There is no relative motion between the steel and fuel, and the relative velocity between the molten fuel and the vapor bubbles is quite low, due to the large drag and inertial forces acting on the bubbles.

As the volume fraction of fuel decreases, the bubbly flow regime is changed to the annular fuel flow regime, with the fuel blanketing the cladding partially or totally and the two-phase sodium/gas mixture flowing at the center of the channel. A stream of solid fuel chunks, interacting with both the two-phase sodium/gas mixture and the liquid fuel streams, can also be present. Molten steel droplets are imbedded in the fuel film. The relative velocities between the fuel and mixture are significantly higher than in the bubbly flow regime.

When molten steel is the dominant component at a certain axial location, the flow regime used can be either an annular or a bubbly steel regime. The bubbly steel flow regime is characterized by the presence of a large amount of molten steel, with droplets of molten fuel, solid fuel and steel chunks, and bubbles of fission gas and sodium vapor imbedded. As the volume fraction of molten steel decreases, the bubbly steel regime is changed to the annular steel flow regime. In this regime, the molten steel totally covers the cladding and the vapor mixture flows at the center of the channel. Molten fuel droplets can be imbedded in the molten steel, and solid fuel and steel chunks can interact with both the vapor mixture and the molten steel. In the Release 1.1 version, a partial annular steel flow regime has been introduced. The steel is assumed to cover the cladding only partially moving in the form of droplets and rivulets. This picture is supported by experimental evidence that indicates that the steel does not wet the cladding. As the local amount of molten steel increases, a full annular steel flow regime can develop. The partial annular steel flow regime is used only when the chunk model is active, i.e. ICHUNK $= 1$. If the chunk model is not used, the full annular steel flow regime is still used. It is expected that, as the validation effort continues, the partial annular steel flow regime will be available, regardless of the chunk model option used.

Because only very limited information is currently available about the fuel-steel flow regime transitions in fuel pin bundles, these transitions are governed in LEVITATE by the fuel-steel volume fraction present at each axial location.

The continuous fuel flow regimes used in LEVITATE have been shown to slow down substantially the fuel dispersal, bringing the results of calculations in close agreement with the experimental data [16-1, 16-11].

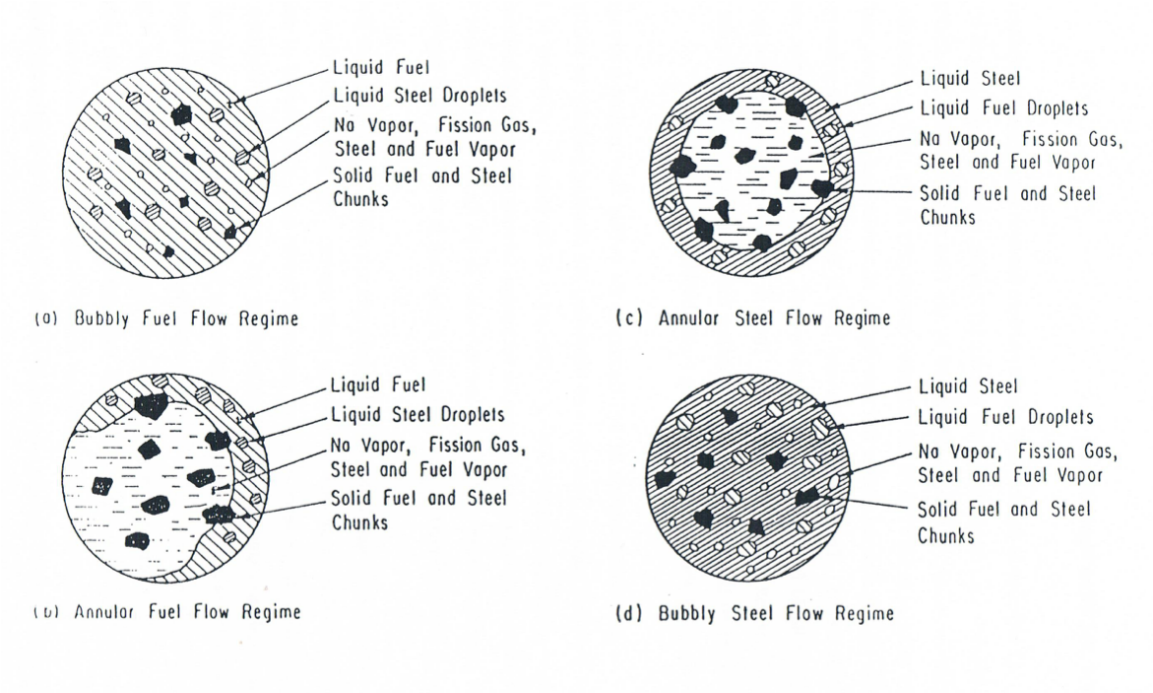


Figure 16.4.1: Continuous Fuel/Steel Regimes Modeled in LEVITATE

16.4.1.5 Mass, Momentum and Energy Exchange (non-convective)

Mass, momentum and energy exchange can take place among various components at any given time and location. All these transfers are strongly dependent on the flow regime prevailing at the time and location considered. The mass transfers are due to phase changes (freezing, melting, vaporization, condensation) or to disruption events, such as the disruption of the solid fuel pins or fuel crust breakup. The mass transfer is largely independent of the local flow regime and the allowable transfers are summarized in Figure 16.4.2.

The mass transfer in any given computational cell, together with the changes in the mass of each component due to convection, determine the mass of each component in the cell. These masses, together with the energy from the previous time step, determine the local flow regime and thus the local configuration. The local energy and momentum transfers are strongly dependent on the local flow regime. The allowable energy transfer paths among various LEVITATE components are illustrated in Figure 16.4.3 to Figure 16.4.6 for various flow regimes. Similarly, the allowable momentum paths are illustrated in Figure 16.4.7 through 16.4-10.

16.4.2 Method of Solution and general Numerical Considerations

16.4.2.1 Variables and Mesh Grid Used in Calculations

The independent variables used in LEVITATE are the axial coordinate, z, and the time, t. Only one spatial coordinate is necessary, as LEVITATE models the subassembly in a one-dimensional geometry. The dependent variables calculated by the hydrodynamic model for each component, are the generalized density ρ', the enthalpy h (or temperature T), and the velocity u. The generalized densities have been introduced in Chapter 14 and, for component k, are defined as:

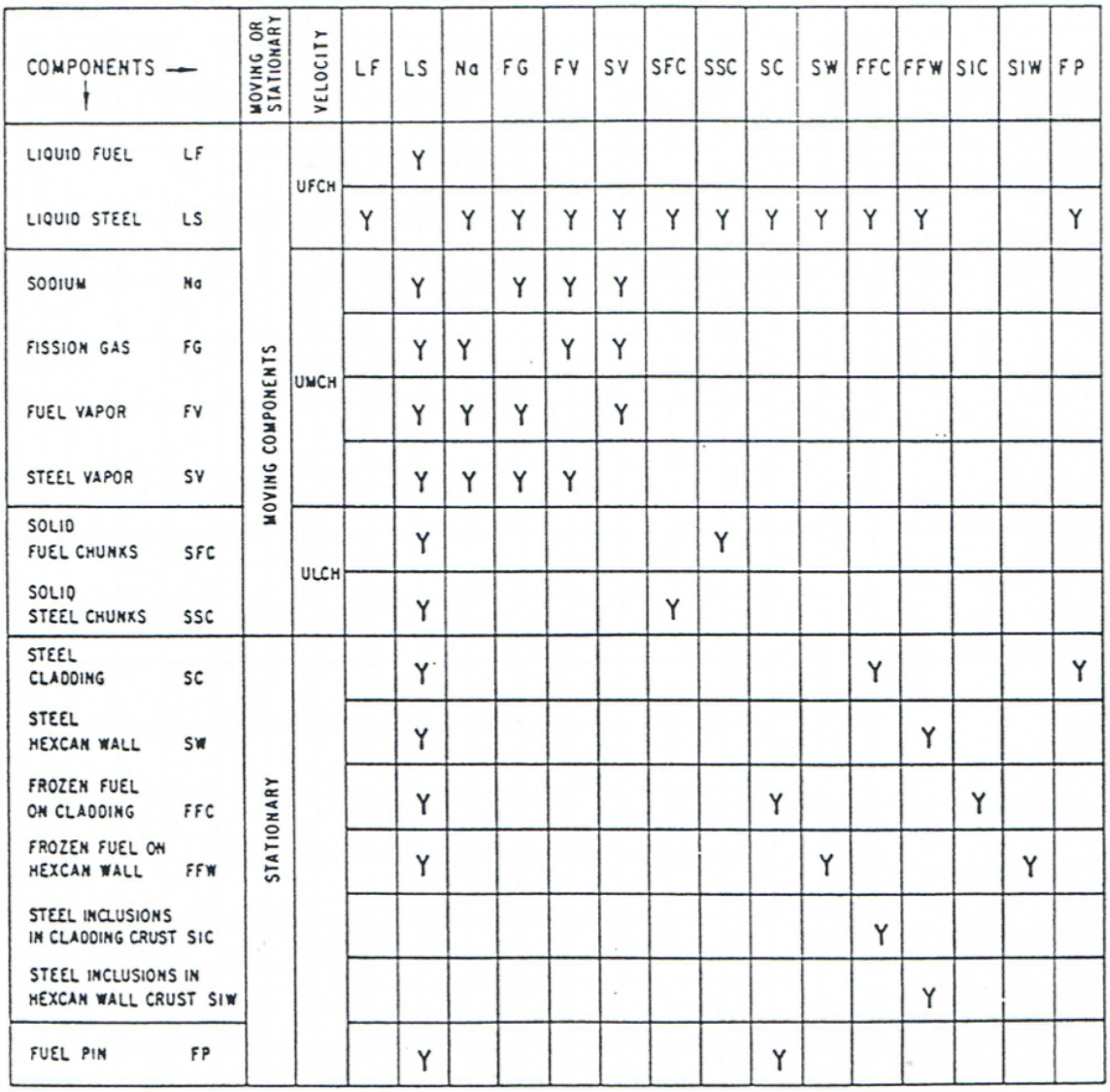
$$\rho'_{\mathrm{k,i}} = \rho_{\mathrm{k,i}} \cdot \frac{A_{\mathrm{k,i}}}{\mathrm{AXMX}} = \rho_{\mathrm{k,i}} \cdot \theta_{\mathrm{k,i}} \tag{16.4-1}$$

COMPONENTS → / ↓		MOVING OR STATIONARY	VELOCITY	LF	LS	Na	FG	FV	SV	SFC	SSC	SC	SW	FFC	FFW	SIC	SIW	FP
LIQUID FUEL	LF	MOVING COMPONENTS	UFCH					Y		Y				Y	Y			
LIQUID STEEL	LS								Y		Y	Y	Y			Y	Y	
SODIUM	Na		UMCH															
FISSION GAS	FG																	
FUEL VAPOR	FV			Y														
STEEL VAPOR	SV				Y													
SOLID FUEL CHUNKS	SFC		ULCH	Y										Y	Y			Y
SOLID STEEL CHUNKS	SSC				Y							Y	Y					
STEEL CLADDING	SC	STATIONARY			Y						Y							
STEEL HEXCAN WALL	SW				Y						Y							
FROZEN FUEL ON CLADDING	FFC			Y						Y								
FROZEN FUEL ON HEXCAN WALL	FFW			Y						Y								
STEEL INCLUSIONS IN CLADDING CRUST	SIC				Y													
STEEL INCLUSIONS IN HEXCAN WALL CRUST	SIW				Y													
FUEL PIN	FP									Y								

Figure 16.4.2: Mass exchange Possibilities among Various LEVITATE Components

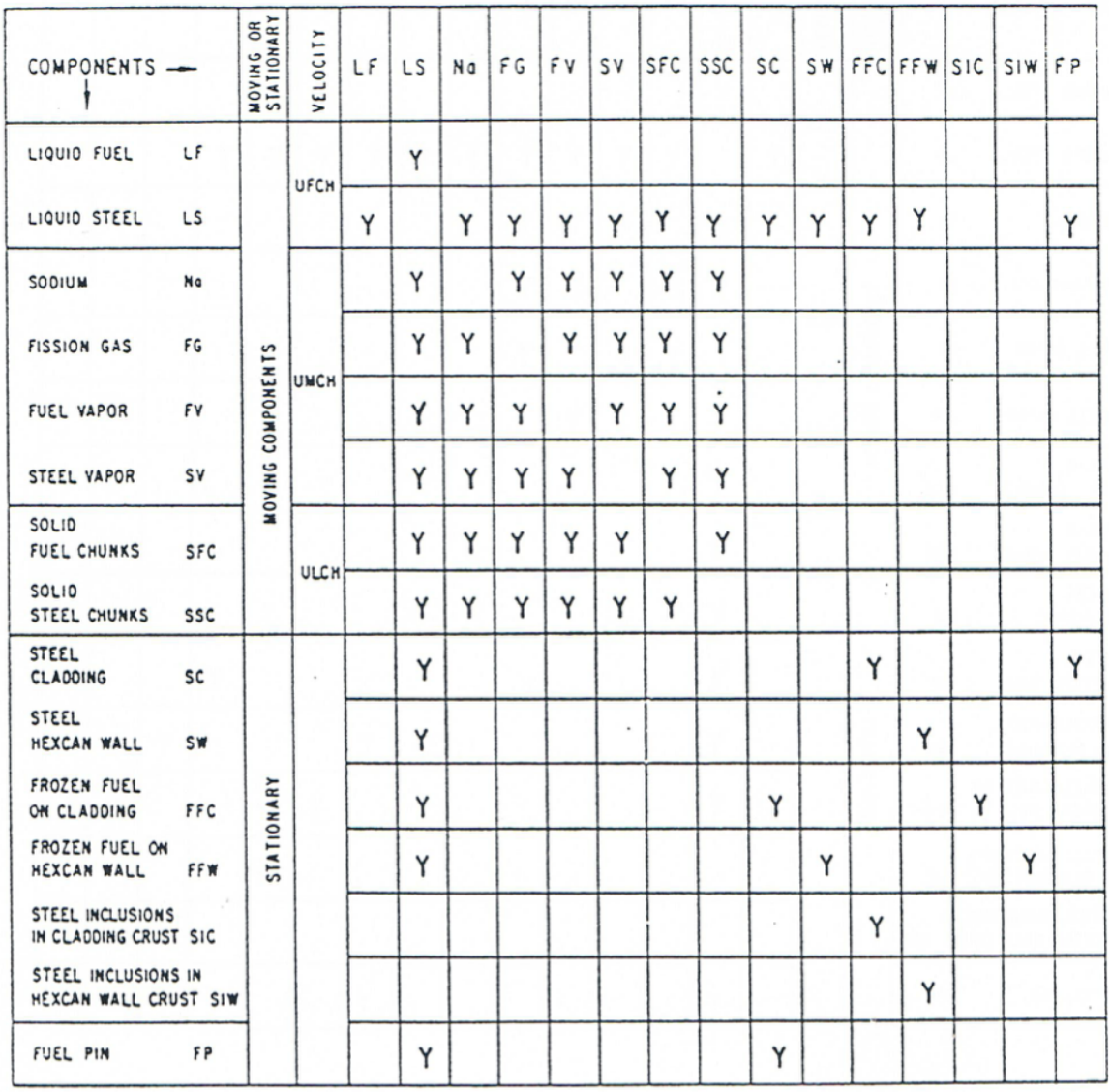
COMPONENTS → / ↓	MOVING OR STATIONARY	VELOCITY	LF	LS	Na	FG	FV	SV	SFC	SSC	SC	SW	FFC	FFW	SIC	SIW	FP
LIQUID FUEL LF	MOVING COMPONENTS	UFCH		Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y			Y
LIQUID STEEL LS			Y														
SODIUM Na		UMCH	Y			Y	Y	Y	Y	Y	Y	Y	Y	Y			Y
FISSION GAS FG			Y		Y		Y	Y	Y	Y	Y	Y	Y	Y			Y
FUEL VAPOR FV			Y		Y	Y		Y	Y	Y	Y	Y	Y	Y			Y
STEEL VAPOR SV			Y		Y	Y	Y		Y	Y	Y	Y	Y	Y			Y
SOLID FUEL CHUNKS SFC		ULCH	Y		Y	Y	Y	Y		Y	Y	Y	Y	Y			Y
SOLID STEEL CHUNKS SSC			Y		Y	Y	Y	Y	Y		Y	Y	Y	Y			Y
STEEL CLADDING SC	STATIONARY		Y		Y	Y	Y	Y	Y	Y			Y				Y
STEEL HEXCAN WALL SW			Y		Y	Y	Y	Y	Y	Y				Y			
FROZEN FUEL ON CLADDING FFC			Y		Y	Y	Y	Y	Y	Y	Y				Y		
FROZEN FUEL ON HEXCAN WALL FFW			Y		Y	Y	Y	Y	Y	Y		Y				Y	
STEEL INCLUSIONS IN CLADDING CRUST SIC													Y				
STEEL INCLUSIONS IN HEXCAN WALL CRUST SIW														Y			
FUEL PIN FP			Y		Y	Y	Y	Y	Y	Y	Y						

Figure 16.4.3: Energy Exchange Possibilities among Various LEVITATE Components, for the Fuel Annular Flow Regime



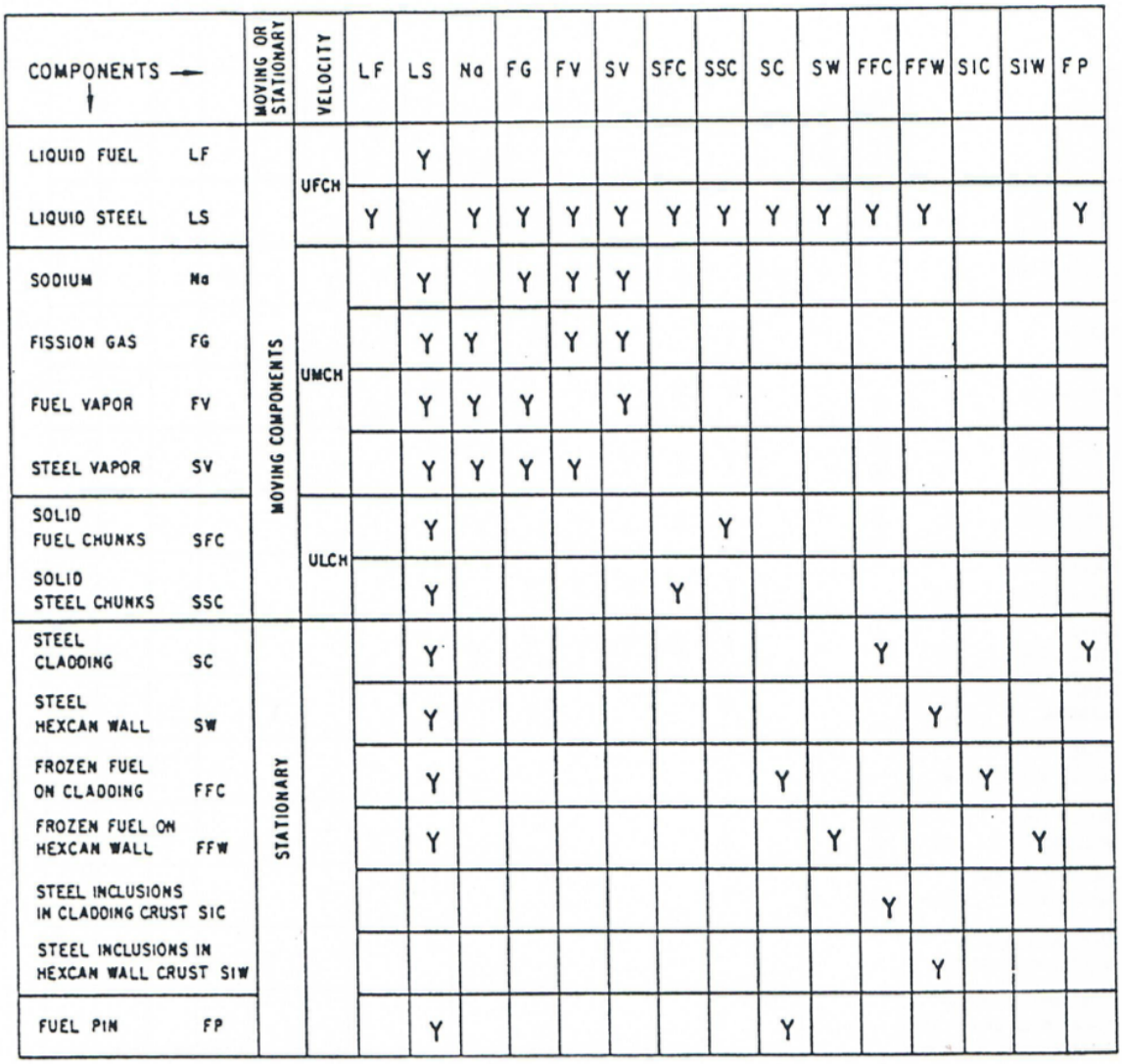
COMPONENTS → / ↓		MOVING OR STATIONARY	VELOCITY	LF	LS	Na	FG	FV	SV	SFC	SSC	SC	SW	FFC	FFW	SIC	SIW	FP
LIQUID FUEL	LF	MOVING COMPONENTS	UFCH		Y													
LIQUID STEEL	LS			Y		Y	Y	Y	Y	Y	Y	Y	Y	Y	Y			Y
SODIUM	Na		UMCH		Y		Y	Y	Y									
FISSION GAS	FG				Y	Y		Y	Y									
FUEL VAPOR	FV				Y	Y	Y		Y									
STEEL VAPOR	SV				Y	Y	Y	Y										
SOLID FUEL CHUNKS	SFC		ULCH		Y						Y							
SOLID STEEL CHUNKS	SSC				Y					Y								
STEEL CLADDING	SC	STATIONARY			Y									Y				Y
STEEL HEXCAN WALL	SW				Y										Y			
FROZEN FUEL ON CLADDING	FFC				Y							Y				Y		
FROZEN FUEL ON HEXCAN WALL	FFW				Y								Y				Y	
STEEL INCLUSIONS IN CLADDING CRUST	SIC													Y				
STEEL INCLUSIONS IN HEXCAN WALL CRUST	SIW														Y			
FUEL PIN	FP				Y							Y						

Figure 16.4.4: Energy Exchange Possibilities among Various LEVITATE Components, for the Fuel Bubbly Flow Regime



COMPONENTS → / ↓	MOVING OR STATIONARY	VELOCITY	LF	LS	Na	FG	FV	SV	SFC	SSC	SC	SW	FFC	FFW	SIC	SIW	FP
LIQUID FUEL LF	MOVING COMPONENTS	UFCH		Y													
LIQUID STEEL LS			Y		Y	Y	Y	Y	Y	Y	Y	Y	Y	Y			Y
SODIUM Na		UMCH		Y		Y	Y	Y	Y	Y							
FISSION GAS FG				Y	Y		Y	Y	Y	Y							
FUEL VAPOR FV				Y	Y	Y		Y	Y	Y							
STEEL VAPOR SV				Y	Y	Y	Y		Y	Y							
SOLID FUEL CHUNKS SFC		ULCH		Y	Y	Y	Y	Y		Y							
SOLID STEEL CHUNKS SSC				Y	Y	Y	Y	Y	Y								
STEEL CLADDING SC	STATIONARY			Y									Y				Y
STEEL HEXCAN WALL SW				Y										Y			
FROZEN FUEL ON CLADDING FFC				Y							Y				Y		
FROZEN FUEL ON HEXCAN WALL FFW				Y								Y				Y	
STEEL INCLUSIONS IN CLADDING CRUST SIC													Y				
STEEL INCLUSIONS IN HEXCAN WALL CRUST SIW														Y			
FUEL PIN FP				Y							Y						

Figure 16.4.5: Energy Exchange Possibilities among Various LEVITATE Components, for the Steel Annular Flow Regime



COMPONENTS → / ↓	MOVING OR STATIONARY	VELOCITY	LF	LS	Na	FG	FV	SV	SFC	SSC	SC	SW	FFC	FFW	SIC	SIW	FP
LIQUID FUEL LF	MOVING COMPONENTS	UFCH		Y													
LIQUID STEEL LS			Y		Y	Y	Y	Y	Y	Y	Y	Y	Y	Y			Y
SODIUM Na		UMCH		Y		Y	Y	Y									
FISSION GAS FG				Y	Y		Y	Y									
FUEL VAPOR FV				Y	Y	Y		Y									
STEEL VAPOR SV				Y	Y	Y	Y										
SOLID FUEL CHUNKS SFC		ULCH		Y						Y							
SOLID STEEL CHUNKS SSC				Y					Y								
STEEL CLADDING SC	STATIONARY			Y									Y				Y
STEEL HEXCAN WALL SW				Y										Y			
FROZEN FUEL ON CLADDING FFC				Y							Y				Y		
FROZEN FUEL ON HEXCAN WALL FFW				Y								Y				Y	
STEEL INCLUSIONS IN CLADDING CRUST SIC													Y				
STEEL INCLUSIONS IN HEXCAN WALL CRUST SIW														Y			
FUEL PIN FP				Y							Y						

Figure 16.4.6: Energy Exchange Possibilities among Various LEVITATE Components, for the Steel Bubbly Flow Regime

LEGEND

Y – ALLOWED MOMENTUM EXCHANGE

X – EQUAL VELOCITIES

COMPONENTS → / ↓		MOVING OR STATIONARY	VELOCITY	LF	LS	Na	FG	FV	SV	SFC	SSC	SC	SW	FFC	FFW	SIC	SIW	FP
LIQUID FUEL	LF	MOVING COMPONENTS	UFCH		X	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y			Y
LIQUID STEEL	LS			X														
SODIUM	Na		UMCH	Y			X	X	X	Y	Y	Y	Y	Y	Y			Y
FISSION GAS	FG			Y		X		X	X	Y	Y	Y	Y	Y	Y			Y
FUEL VAPOR	FV			Y		X	X		X	Y	Y	Y	Y	Y	Y			Y
STEEL VAPOR	SV			Y		X	X	X		Y	Y	Y	Y	Y	Y			Y
SOLID FUEL CHUNKS	SFC		ULCH	Y		Y	Y	Y	Y		X	Y	Y	Y	Y			Y
SOLID STEEL CHUNKS	SSC			Y		Y	Y	Y	Y	X		Y	Y	Y	Y			Y
STEEL CLADDING	SC	STATIONARY																
STEEL HEXCAN WALL	SW																	
FROZEN FUEL ON CLADDING	FFC																	
FROZEN FUEL ON HEXCAN WALL	FFW																	
STEEL INCLUSIONS IN CLADDING CRUST	SIC																	
STEEL INCLUSIONS IN HEXCAN WALL CRUST	SIW																	
FUEL PIN	FP																	

Figure 16.4.7: Momentum Exchange Possibilities among Various LEVITATE Components, for the Fuel Annular Flow Regime

LEGEND

Y – ALLOWED MOMENTUM EXCHANGE

X – EQUAL VELOCITIES

COMPONENTS → / ↓		MOVING OR STATIONARY	VELOCITY	LF	LS	Na	FG	FV	SV	SFC	SSC	SC	SW	FFC	FFW	SIC	SIW	FP
LIQUID FUEL	LF	MOVING COMPONENTS	UFCH		X	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y		Y
LIQUID STEEL	LS			X														
SODIUM	Na		UMCH	Y			X	X	X									
FISSION GAS	FG			Y		X		X	X									
FUEL VAPOR	FV			Y		X	X		X									
STEEL VAPOR	SV			Y		X	X	X										
SOLID FUEL CHUNKS	SFC		ULCH	Y							X							
SOLID STEEL CHUNKS	SSC			Y						X								
STEEL CLADDING	SC	STATIONARY																
STEEL HEXCAN WALL	SW																	
FROZEN FUEL ON CLADDING	FFC																	
FROZEN FUEL ON HEXCAN WALL	FFW																	
STEEL INCLUSIONS IN CLADDING CRUST	SIC																	
STEEL INCLUSIONS IN HEXCAN WALL CRUST	SIW																	
FUEL PIN	FP																	

Figure 16.4.8: Momentum Exchange Possibilities among Various LEVITATE Components, for the fuel Bubbly Flow Regime

LEGEND

Y – ALLOWED MOMENTUM EXCHANGE
X – EQUAL VELOCITIES

COMPONENTS → / ↓	MOVING OR STATIONARY	VELOCITY	LF	LS	Na	FG	FV	SV	SFC	SSC	SC	SW	FFC	FFW	SIC	SIW	FP
LIQUID FUEL LF	MOVING COMPONENTS	UFCH		X													
LIQUID STEEL LS			X		Y	Y	Y	Y	Y	Y	Y	Y	Y	Y			Y
SODIUM Na		UMCH		Y		X	X	X	Y	Y							
FISSION GAS FG				Y	X		X	X	Y	Y							
FUEL VAPOR FV				Y	X	X		X	Y	Y							
STEEL VAPOR SV				Y	X	X	X		Y	Y							
SOLID FUEL CHUNKS SFC		ULCH		Y	Y	Y	Y	Y		X							
SOLID STEEL CHUNKS SSC				Y	Y	Y	Y	Y	X								
STEEL CLADDING SC	STATIONARY																
STEEL HEXCAN WALL SW																	
FROZEN FUEL ON CLADDING FFC																	
FROZEN FUEL ON HEXCAN WALL FFW																	
STEEL INCLUSIONS IN CLADDING CRUST SIC																	
STEEL INCLUSIONS IN HEXCAN WALL CRUST SIW																	
FUEL PIN FP																	

Figure 16.4.9: Momentum Exchange Possibilities among Various LEVITATE Components, for the Steel Annular Flow Regime

LEGEND

Y – ALLOWED MOMENTUM EXCHANGE

X – EQUAL VELOCITIES

COMPONENTS → / ↓		MOVING OR STATIONARY	VELOCITY	LF	LS	Na	FG	FV	SV	SFC	SSC	SC	SW	FFC	FFW	SIC	SIW	FP
LIQUID FUEL	LF	MOVING COMPONENTS	UFCH		X													
LIQUID STEEL	LS			X		Y	Y	Y	Y	Y	Y	Y	Y	Y	Y			Y
SODIUM	Na		UMCH		Y		X	X	X									
FISSION GAS	FG				Y	X		X	X									
FUEL VAPOR	FV				Y	X	X		X									
STEEL VAPOR	SV				Y	X	X	X										
SOLID FUEL CHUNKS	SFC		ULCH		Y						X							
SOLID STEEL CHUNKS	SSC				Y					X								
STEEL CLADDING	SC	STATIONARY																
STEEL HEXCAN WALL	SW																	
FROZEN FUEL ON CLADDING	FFC																	
FROZEN FUEL ON HEXCAN WALL	FFW																	
STEEL INCLUSIONS IN CLADDING CRUST	SIC																	
STEEL INCLUSIONS IN HEXCAN WALL CRUST	SIW																	
FUEL PIN	FP																	

Figure 16.4.10: Momentum Exchange Possibilities among Various LEVITATE Components, for the Steel Bubbly Flow Regime

where

$\rho_{k,i}$ = physical density of component k at location i

$A_{k,i}$ = cross sectional area occupied by component k at location i

AXMX = reference area

$\theta_{k,i}$ = generalized (area) fraction of component k at location i

The mass, energy and momentum partial differential conservation equations are solved using an Eulerian finite difference semi-explicit formulation, as explained below. The mesh grid used for the finite difference formulation is presented in Figure 16.4.11. As indicated in this figure, the densities and enthalpies are defined at the center of each cell, while the velocities are defined at the boundaries. Because of the highly irregular geometry treated by LEVITATE, special attention was necessary for the treatment of abrupt area changes [16-12]. Fuel velocities are defined at each cell boundary, with u_i being the velocity just before boundary i, and u''_i the velocity just after that boundary. The terms "before" and "after" are used in relation to the positive direction of the axial coordinate z.

In order to reduce the numerical diffusion, characteristic of Eulerian numerical schemes, the boundaries of each region containing a certain component are tracked separately as they move through the Eulerian cells.

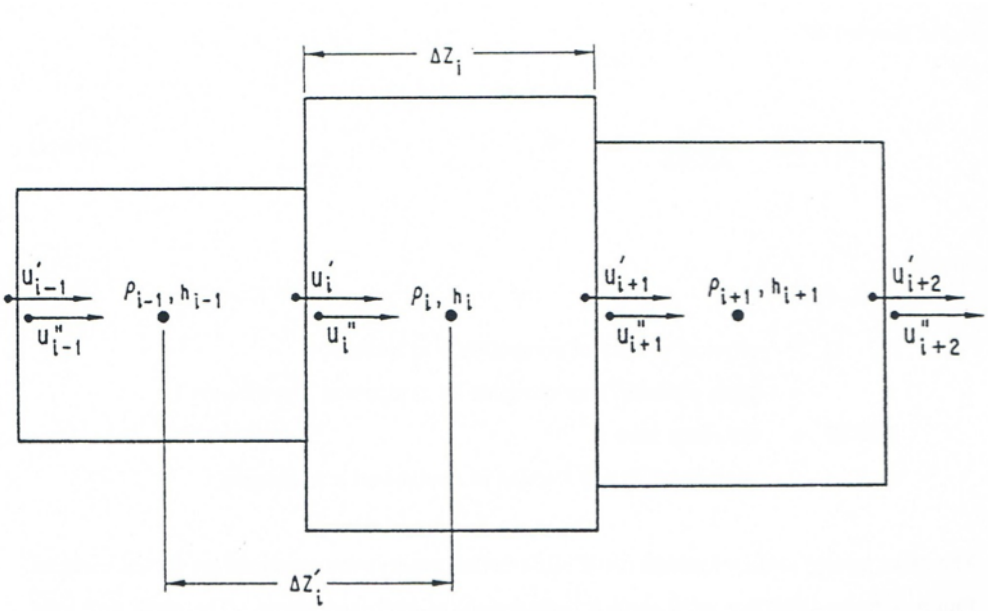


Figure 16.4.11: Mesh Grid used in the Channel Hydrodynamic Model

16.4.2.2 Description of the Method of Solution and Logic Flow

A simplified modular chart of the LEVITATE model is presented in Figure 16.1.5. The calculation begins by calculating the changes in the position of material boundaries during the current time step. The new interface positions, at the end of the time step, are calculated for all components, except sodium in the LEIF (**LE**VITATE **I**NTER**F**ACE) routine. The position of the sodium slugs, which determine the boundaries of the sodium region, is calculated in the LEREZO (**LE**VITATE **REZO**NING) routine. This routine can add (or remove) nodes to the LEVITATE compressible region as the sodium slugs move out of (or into) the channel. The mass conservation equation is solved next, for all components and all axial locations. Each equation is solved explicitly, i.e., the convective fluxes are based on the generalized densities present in each cell at the beginning of the time step.

These calculations are performed in the LEMACO (**LE**VITATE **MA**SS **CO**NSERVATION) routine. The new densities are then used in the LEVOFR (**LE**VITATE **VO**LUME **FR**ACTION) routine to determine the volume fraction of each component at each axial location. Using these volume fractions, the LEVOFR routine also determines the flow regime in each axial cell. This flow regime will be assumed to exist in the cell for the duration of the time step. It is worth noting that this is an "implicit type" assumption as the flow regimes are based on the densities calculated at the end of the time step. The next routine called is LEGEOM (LEVITATE GEOMETRY) which determines the geometrical characteristics defining each local configuration. This routine will change the thickness of the fuel crust as necessary or determine the fraction of the cladding circumference covered by the liquid fuel in the partial annular flow regime. In general, LEGEOM calculates the area of contact between various components in various flow regimes. These areas will be used later in calculating the energy and momentum transfer between various components.

The next routine called is LETRAN (**LE**VITATE **TRAN**SFER) which calculates the heat-transfer and friction coefficients for all axial locations and among all components that are in direct contact. Thus, the code will use the flow regimes present at a certain location to determine which heat transfer and friction coefficients have to be calculated. The allowable exchanges for each flow regime and each component have already been presented in Figure 16.4.3 through Figure 16.4.10. Once contact areas and the corresponding heat-transfer coefficients have been calculated, the energy conservation equation can be solved for all components. The LESOEN (**LE**VITATE **SO**LID, LIQUID AND STATIONARY **EN**ERGY EQUATION) routine is called to solve the energy conservation equations for the fuel and steel channels, liquid fuel, liquid steel, stationary cladding, hexcan wall, frozen fuel on the cladding and frozen fuel on the hexcan wall. All equations are solved explicitly, i.e., the convective fluxes are based on beginning of time step densities, thus allowing the axial decoupling of the equations. The energy equation for sodium (two-phase or single-phase vapor) and fission gas is solved in LENAEN (**LE**VITATE SODIUM-**NA EN**ERGY). The energy equations for fuel and steel vapor are solved in LEFUVA (**LE**VITATE **FU**EL **VA**POR ENERGY) and LESEVA (**LE**VITATE **S**TEEL **VA**POR ENERGY), respectively. The new temperatures calculated in LENAEN, LEFUVA and LESEVA are used to determine the new pressure of each of the compressible components and thus the total new pressure. The hydrodynamic in-pin model is then used to advance the in-pin solution in the LElPIN and LE2PIN routines. These routines interact with the channel model via the fuel injection process, which is modeled in the LElPIN routine. Molten fuel and fission gas are ejected from the cavity into the channel, leading to changes in the local pressure. The momentum equation for each of the three velocity fields is then solved in the routine LEMOCO (**LE**VITATE **MO**MENTUM **CO**NSERVATION). The method of solution is still explicit and the equations are uncoupled axially, but the equations for all three fields are solved simultaneously rather than independently, as was done in the mass and energy equations. Also it is important to note that the pressures used in the momentum equation are the pressures at the end of the time step. The routine called next, LELUME (**LE**VITATE CHUNK-**LU ME**LTING), calculates the melting and the size changes of the solid fuel/steel chunks at all axial locations. The routine LEFREZ (**LE**VITATE **FREEZ**ING AND MELTING) then models a series of important processes, such as fuel/steel freezing and crust formation, fuel/steel chunk formation, fuel crust remelting and breakup. Next called is the routine LEABLA (**LE**VITATE **ABLA**TION) which calculates the gradual melting and ablation of the cladding and hexcan wall. The routine LEDISR (**LE**VITATE **DISR**UPTION) performs the disruption of the fuel pin whenever a disrupted node is predicted. The disruption, which can occur in one or more nodes in any time step, leads to changes in geometry, mass, energy and pressure for various components present in the respective cell. Finally, the routine LESRME (**LE**VITATE **STRU**CTURE **ME**LTING) calculates the rupture of the hexcan wall, due to melting and/or pressure burst effects.

16.4.3 Differential Equations and Finite Difference Equations

16.4.3.1 Mass Conservation Equations

This section describes the solution of the mass conservation equations for a generic component k, as performed in the routine LEMACO. The region where the k component is present has been previously updated in the routines LEIF and LEREZO, and is defined by two integers $i_{\text{k,BT}}$ and $i_{\text{k,TP}}$, indicating the number of the bottom and top node of the region, respectively. A mass conservation equation is solved for each axial node i, with $i_{\text{k,BT}} \leq i \leq i_{\text{k,TP}}$. The original mass conservation is written as:

$$\frac{\partial}{\partial \text{t}} \left(\rho_{\text{k,i}} \cdot A_{\text{k,i}} \cdot \Delta z_{\text{i}}\right) + \left[(\rho A u)_{\text{k,i+1/2}} - (\rho A u)_{\text{k,i-1/2}}\right] = 0 \tag{16.4-2}$$

No source terms are considered in LEMACO, as all phase changes and injections are treated in separate routines, which will be described later.

After dividing by AXMX $\cdot \Delta z_{\text{i}}$ and using the definition of generalized densities, we obtain:

$$\frac{\partial}{\partial \text{t}} \rho'_{\text{k,i}} = -\left[(\rho' u)_{\text{k,i+1/2}} - (\rho' u)_{\text{k,i-1/2}}\right] \cdot \frac{1}{\Delta z_{\text{i}}} \tag{16.4-3}$$

Finally, after integrating over the length of the time step, Δt, we obtain:

$$\Delta \rho'_{\text{k,i}} = -\left[(\rho' u)_{\text{k,i+1/2}} - (\rho' u)_{\text{k,i-1/2}}\right] \cdot \frac{\Delta t}{\Delta z_{\text{i}}} \tag{16.4-4}$$

or

$$\rho'^{n+1}_{\text{k,i}} = \rho'^{n}_{\text{k,i}} - \left[(\rho' u)_{\text{k,i+1/2}} - (\rho' u)_{\text{k,i-1/2}}\right] \cdot \frac{\Delta t}{\Delta z_{\text{i}}} \tag{16.4-5}$$

In relation to the use of the superscripts, it is noted that whenever a time-dependent quantity such as $(\rho' u)$ is written without a superscript, its value is evaluated at the beginning of the computational time step.

The quantity $(\rho' u)_{\text{k,i-1/2}}$ is the mass convective flux of component k at the boundary $i - 1/2$ and is calculated using an upstream differencing approach:

If $u'_{\text{k,i}} \geq 0$,

$$(\rho' u)_{\text{k,i-1/2}} = \rho'_{\text{k,i-1}} \cdot u'_{\text{k,i}} \tag{16.4-6}$$

If $u'_{\text{k,i}} < 0$, then

$$(\rho' u)_{\text{k,i-1/2}} = \rho'_{\text{k,i}} \cdot u''_{\text{k,i}} \tag{16.4-7}$$

The velocities $u'_{\text{k,i}}$ and $u''_{\text{k,i}}$ are correlated by the mass continuity equation across boundary $i - 1/2$:

$$u'_{\text{k,i}} \rho_{\text{k,i-1}} A_{\text{k,i-1}} = u''_{\text{k,i}} \cdot \rho_{\text{k,i}} \cdot A_{\text{k,i}} \tag{16.4-8}$$

The solution of the mass equation is illustrated below for the molten fuel component. The molten fuel region extends from $i_{\text{fu,BT}} =$ IFFUBT to $i_{\text{fu,TP}} =$ IFFUTP. The first step in the solution is to calculate the convective fuel fluxes at all internal cell boundaries. It is important to note that the convective fluxes defined at boundary $i - 1/2$ have the subscript i, in the code, i.e., COFUCH (I). Similarly, those defined at the boundary $i + 1/2$ have the subscript $i + 1$.

Note that the velocity $u''_{\text{k,i}}$ is stored as UFCH (I), and the velocity $u'_{\text{k,i}}$ used in the previous equations is calculated as:

$$u'_{\text{fu,i}} = u''_{\text{fu,i}} \cdot \text{CCFU}\,(I) \tag{16.4-9}$$

where the coefficient CCFU (I) is defined by the Eq. (16.4-8) as:

$$\text{CCFU}\,(I) = \frac{u'_{\text{fu,i}}}{u''_{\text{fu,i}}} = \frac{\rho_{\text{fu,i}} \cdot A_{\text{fu,i}}}{\rho_{\text{fu,i-1}} \cdot A_{\text{fu,i-1}}} = \frac{\rho'_{\text{fu,i}}}{\rho'_{\text{fu,i-1}}} \tag{16.4-10}$$

The convective fluxes through the boundaries of the fuel region are set to zero, as no fuel is allowed to cross these boundaries in the LEMACO routine.

$$(\rho' u)_{\text{fu,IFFUBT}} = 0 \tag{16.4-11}$$

$$(\rho' u)_{\text{fu,IFFUTP+1}} = 0 \tag{16.4-12}$$

To preserve the accuracy of results in the boundary cells a correction is applied to the convective fluxes through all boundaries IFFUBT $+ 1$ and IFFUTP whenever the corresponding interface (bottom and top) crosses that boundary during the time step. This correction is explained below for the case when the top fuel boundary crosses the cell boundary. The location of the interface at the end of the time step has already been calculated in the routine LEIF and is shown in Figure 16.4.12. The value of $(\rho' u)_{\text{fu,IFFUTP}}$ is calculated according to Eq. (16.4-8) and then used in Eq. (16.4-5). The implicit assumption in Eq. (16.4-5) is that the value of the convective fluxes is constant over the length of the time step. However, as shown in Figure 16.4.12 the original location of the fuel interface was FUIFTP^0 (1), and, before the interface reaches the cell boundary ZC(IFFUTP), the flux across the boundary is zero. The fraction of Δt during which the convective flux is present is:

$$F_{\text{correction}} = \frac{\text{FUIFTP}\,(1) - \text{ZC}\,(\text{IFFUTP})}{\text{FUIFTP}^0\,(1) - \text{FUIFTP}\,(1)} \tag{16.4-13}$$

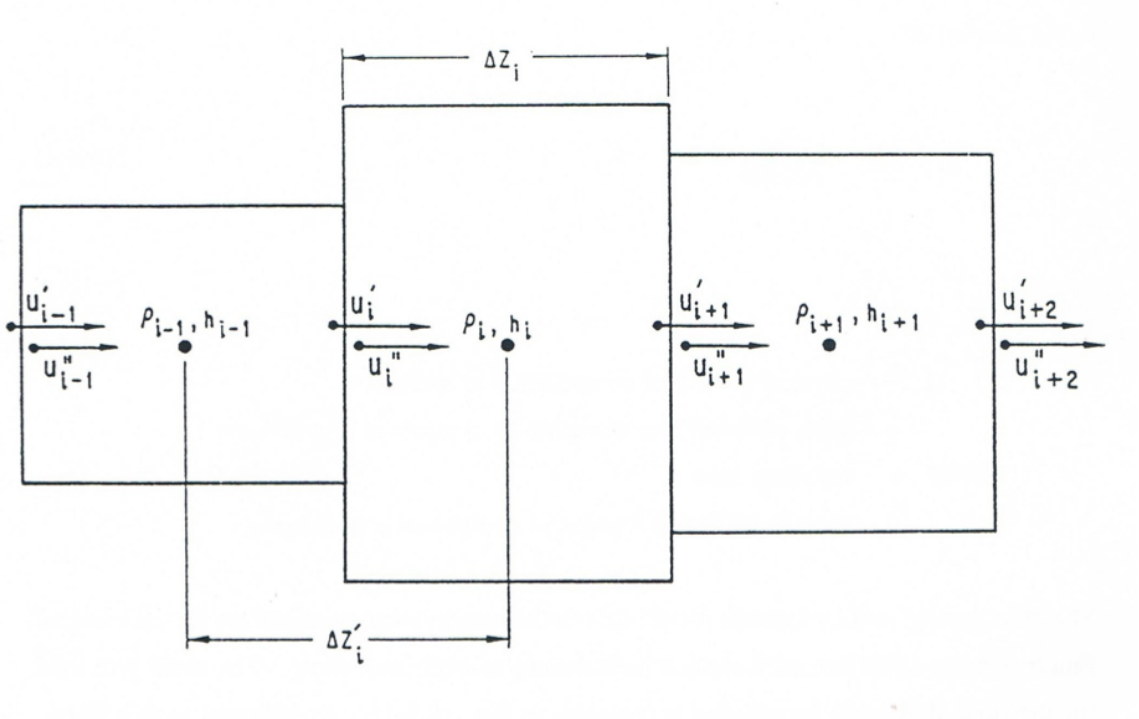


Figure 16.4.12: Figure 16.4.12: Correction for Material Boundary Crossing a Cell Boundary

In order to maintain the form of Eq. (16.4-5), rather than changing the value of Δt associated with $(\rho' u)_{\text{fu,IFFUTP}}$ a corrected flux is defined, such that the product flux * time is correct:

$$(\rho' u)_{\text{fu,IFFUTP}} = (\rho' u)_{\text{fu,IFFUTP}} \cdot F_{\text{correction}} \tag{16.4-14}$$

A similar correction is applied for the bottom boundary.

Similar mass conservation equations are solved for all other LEVITATE components, i.e. molten steel, fuel and steel chunks, fission gas, sodium, steel vapor, fuel vapor, fission gas still present in the chunks and fission gas present in the molten fuel.

16.4.3.2 Liquid Fuel Energy Conservation Equations

The liquid fuel energy equation is solved in the routine LESOEN, for all cells in the molten fuel region, i.e., IFFUBT $\leq I \leq$ IFFUTP. We begin with the energy equation in conservation form:

$$\begin{aligned}\frac{\partial}{\partial \mathrm{t}}\left(\rho_{\mathrm{fu,i}} \cdot h_{\mathrm{fu,i}} \cdot A_{\mathrm{fu,i}} \cdot \Delta z_{\mathrm{i}}\right) + \left[(\rho A h u)_{\mathrm{i+1/2}} - (\rho A h u)_{\mathrm{i\text{-}1/2}}\right] \\ = Q_{\mathrm{fu,i}} \cdot \rho_{\mathrm{fu,i}} \cdot A_{\mathrm{fu,i}} \cdot \Delta z_{\mathrm{i}} - \sum_{\mathrm{j}} H_{\mathrm{fu,j,i}} \cdot A_{\mathrm{fu,j,i}} \cdot \Delta T_{\mathrm{fu,j,i}}\end{aligned} \tag{16.4-15}$$

where:

$Q_{\mathrm{fu,i}}$ = fission power source in cell i [J/(kg-s)]

$H_{\mathrm{fu,j,i}}$ = heat-transfer coefficient between fuel and component j in cell i
[J/(m^2 - s - K)]

$A_{\mathrm{fu,j,i}}$ = heat-transfer area between fuel and component j in cell i [m^2]

After dividing by AXMX $\cdot \Delta z_{\mathrm{i}}$ and using the definition of generalized densities, we obtain:

$$\begin{aligned}\frac{\partial}{\partial \mathrm{t}}\left(\rho'_{\mathrm{fu,i}} \cdot h_{\mathrm{fu,i}}\right) = &- \left[(\rho' h u)_{\mathrm{i+1/2,fu}} - (\rho' h u)_{\mathrm{i\text{-}1/2,fu}}\right] \cdot \frac{1}{\Delta z_{\mathrm{i}}} \\ &+ Q_{\mathrm{fu,i}} \cdot \rho'_{\mathrm{fu,i}} - \sum_{\mathrm{j}} H'_{\mathrm{fu,j,i}} \cdot \Delta T_{\mathrm{fu,j,i}}\end{aligned} \tag{16.4-16}$$

where

$$H'_{\mathrm{fu,j,i}} = H_{\mathrm{fu,j,i}} \cdot \frac{A_{\mathrm{fu,j,i}}}{\mathrm{AXMX} \cdot \Delta z_{\mathrm{i}}} \tag{16.4-17}$$

Integrating over Δt and using the identity:

$$\Delta\left(\rho' h\right) = \rho'^{n+1} \cdot \Delta h + h^n \cdot \Delta \rho' \tag{16.4-18}$$

we obtain:

$$\begin{aligned}\rho'^{n+1}_{\mathrm{fu,i}} \Delta h_{\mathrm{fu,i}} = &- \left[(\rho' h u)_{\mathrm{i+1/2,fu}} - (\rho' h u)_{\mathrm{i\text{-}1/2,fu}}\right] \cdot \frac{\Delta t}{\Delta z_{\mathrm{i}}} \\ &- h^n_{\mathrm{fu,i}} \cdot \Delta \rho'_{\mathrm{fu,i}} + Q'_{\mathrm{fu,i}} \cdot \rho'_{\mathrm{fu,i}} \cdot \Delta t \\ &- \sum_{\mathrm{j}} H'_{\mathrm{fu,j,i}} \cdot \Delta T_{\mathrm{fu,j,i}} \cdot \Delta t\end{aligned} \tag{16.4-19}$$

Finally, dividing Eq. (16.4-19) by $\rho'^{n+1}_{\mathrm{fu,i}}$ and expressing $\Delta \rho'_{\mathrm{fu,i}}$ as:

$$\Delta \rho'_{\mathrm{fu,i}} = - \left[(\rho' u)_{\mathrm{fu,i+1/2}} - (\rho' u)_{\mathrm{fu,i\text{-}1/2}}\right] \cdot \frac{\Delta t}{\Delta z_{\mathrm{i}}} \tag{16.4-20}$$

from the continuity Eq. (16.4-4), we obtain the change in the fuel enthalpy:

$$\begin{aligned}\Delta h_{\text{fu,i}} = \Bigg\{ &- \left[(\rho' hu)_{\text{fu,i+1/2}} - (\rho' hu)_{\text{fu,i-1/2}}\right] \cdot \frac{1}{\Delta z_{\text{i}}} \\ &+ h^{n}_{\text{fu,i}} \left[(\rho' u)_{\text{fu,i+1/2}} - (\rho' u)_{\text{fu,i-1/2}}\right] \cdot \frac{1}{\Delta z_{\text{i}}} \\ &+ Q_{\text{fu,i}} \cdot \rho'_{\text{fu,i}} - \sum_{\text{j}} H'_{\text{fu,j,i}} \cdot \Delta T_{\text{fu,j,i}} \Bigg\} \cdot \frac{\Delta t}{\rho'^{\,n+1}_{\text{fu,i}}}\end{aligned} \tag{16.4-21}$$

The new fuel enthalpy is obtained as:

$$h^{n+1}_{\text{fu,i}} = h^{n}_{\text{fu,i}} + \Delta h_{\text{fu,i}} \tag{16.4-22}$$

The energy convective terms in Eq. (16.4-21) are calculated using an upstream differencing approach, i.e.;

$$(\rho' hu)_{\text{fu,i-1/2}} = \begin{cases} \rho'_{\text{fu,i-1}} \cdot h_{\text{fu,i-1}} \cdot u'_{\text{fu,i}} & \text{if } u'_{\text{fu,i}} \geq 0 \\ \rho'_{\text{fu,i}} \cdot h_{\text{fu,i}} \cdot u''_{\text{fu,i}} & \text{if } u'_{\text{fu,i}} < 0 \end{cases} \tag{16.4-23}$$

These fluxes are based on the fuel densities, enthalpies and velocities at the of the time step. They are calculated in the routine LEMACO, before the calculation of the new densities, and stored in the array COFUOS (I). The sum of the heat-transfer contributions, $\sum_{\text{j}} H'_{\text{fu,j,i}} \cdot \Delta T_{\text{fu,j,i}}$ is presented in detail in Eq. (16.4-24).

In the code, Eq. (16.4-21) is written as:

$$\begin{aligned}\text{DEGEOS} = [&-\text{COFUOA} + \text{COFUOB} + \text{HSFU}\,(I) * \text{DEFUCH}\,(I) \\ &-\text{HTFUNA}\,(I) * (\text{TEFUOS}\,(I) - \text{TENA}\,(I)) \\ &-\text{HTFUCL}\,(I) * (\text{TEFUOS}\,(I) - \text{TECLOS}\,(I)) \\ &-\text{HTFUFL}\,(I) * (\text{TEFUOS}\,(I) - \text{TELUCH}\,(I)) \\ &-\text{HTFUSL}\,(I) * (\text{TEFUOS}\,(I) - \text{TESELU}\,(I)) \\ &-\text{HTFUSR}\,(I) * (\text{TEFUOS}\,(I) - \text{TESROS}\,(I)) \\ &-\text{HTSEFU}\,(I) * (\text{TEFUOS}\,(I) - \text{TESECH}\,(I))] * \text{DTPLU}/\text{DEFUCH}\,(I)\end{aligned} \tag{16.4-24}$$

where

$$\text{COFUOA} = \left[(\rho' hu)_{\text{fu,i+1/2}} - (\rho' hu)_{\text{fu,i-1/2}}\right] \cdot \frac{1}{\Delta z_{\text{i}}} \tag{16.4-25}$$

$$\text{COFUOB} = h^{n}_{\text{fu,i}} \cdot \left[(\rho' u)_{\text{fu,i+1/2}} - (\rho' u)_{\text{fu,i-1/2}}\right] \cdot \frac{1}{\Delta z_{\text{i}}} \tag{16.4-26}$$

HTFUNA $(I) = H'_{\text{fu,mi,i}}$ = generalized heat-transfer coefficient between fuel and gas mixture in cell i

HTFUCL $(I) = H'_{\text{fu,cl,i}}$ = generalized heat-transfer coefficient between fuel and cladding in cell i

HTFUFL $(I) = H'_{\text{fu,fl,i}}$ = generalized heat-transfer coefficient between fuel and fuel chunks in cell i

HTFUSL $(I) = H'_{\text{fu,sl,i}}$ = generalized heat-transfer coefficient between fuel and steel chunks in cell i

HTFUSR $(I) = H'_{\text{fu,sr,i}}$ = generalized heat-transfer coefficient between fuel and hexcan wall in cell i

HTSEFU $(I) = H'_{\text{fu,se,i}}$ = generalized heat-transfer coefficient between fuel and molten steel in cell i

The generalized transfer coefficients $H'_{\text{fu,j,i}}$ are related to the heat-transfer coefficients $H_{\text{fu,j,i}}$ and the transfer areas $A_{\text{fu,j,i}}$ by Eq. (16.4-17). These transfer coefficients are calculated prior to the energy equation solution in the routine LETRAN (**LE**VITATE **TRAN**SFER). Depending on the flow regime, some of those coefficients can be zero, as illustrated in the decision arrays Figure 16.4.3 through Figure 16.4.6.

After the energy change is calculated, and the new energy is obtained according to Eq. (16.4-22), the temperature of the fuel is calculated by using an external function, TEFUEG, which uses the enthalpy as argument

$$T_{\text{fu,i}}^{n+1} = T\left(h_{\text{fu,i}}^{n+1}\right) \tag{16.4-27}$$

Several checks are performed on the final temperature in order to avoid numerical difficulties. Thus, if only small amounts of fuel are present (fuel volume less than .1% of the cell volume) the temperature of the fuel is set equal to the cladding temperature or, if the fuel is surrounded by molten steel, to the molten steel temperature.

The condensation/vaporization energy sources/sinks are not included here. They will be introduced later in this chapter, when presenting the energy conservation calculation for fuel vapor.

16.4.3.3 Liquid Steel Energy Conservation Equation

The liquid steel energy equation is solved in the routine LESOEN, for all cells in the molten steel region, i.e. IFSEBT $\leq I \leq$ IFSETP. The energy conservation equation is similar to the molten fuel equation, but does not include a fission source term. Thus, the change in the steel enthalpy is:

$$h_{\text{se,i}} = \left\{ -\left[(\rho' hu)_{\text{se,i+1/2}} - (\rho' hu)_{\text{se,i-1/2}}\right] \cdot \frac{1}{z_{\text{i}}} + h_{\text{se,i}}^{n}\left[(\rho' u)_{\text{se,i+1/2}} - (\rho' u)_{\text{se,i-1/2}}\right] \cdot \frac{1}{z_{\text{i}}} - \sum_{\text{j}} H'_{\text{se,j,i}} \cdot T_{\text{se,j,i}} \right\} \cdot \frac{t}{\rho'^{n+1}_{\text{se,i}}} \tag{16.4-28}$$

The new steel enthalpy is obtained as:

$$h_{\text{se,i}}^{n+1} = h_{\text{se,i}}^{n} + \Delta h_{\text{se,i}} \tag{16.4-29}$$

The energy convective terms in Eq. (16.4-28) are calculated using an upstream differencing approach, similar to Eq. (16.4-21). They are calculated in routine LEMACO and stored under the name COSEOS (I). These fluxes are based on steel densities enthalpies and velocities at the beginning of the time step.

If $u'_{\text{se,i}} \geq 0$,

$$(\rho' hu)_{\text{se,i-1/2}} = \rho'_{\text{se,i-1}} \cdot h_{\text{se,i-1}} \cdot u'_{\text{se,i}} \tag{16.4-30}$$

If $u'_{\text{se,i}} < 0$, then

$$(\rho' hu)_{\text{se,i-1/2}} = \rho'_{\text{se,i}} \cdot h_{\text{se,i}} \cdot u''_{\text{se,i}} \tag{16.4-31}$$

As previously explained, the molten steel and molten fuel share the same velocity field. Thus $u'_{\text{se,i}}$ and $u'_{\text{fu,i}}$ have the same value, stored in the array UFCH (I).

In the code, Eq. (16.4-28) is written as:

$$\begin{aligned} \text{DEEGSE} = [&-\text{COSEOA} + \text{COSEOB} \\ &-\text{HTSEFU}\,(I) * (\text{TESECH}\,(I) - \text{TEFUOL}\,(I)) \\ &-\text{HTSECL}\,(I) * (\text{TESECH}\,(I) - \text{TECLOS}\,(I)) \\ &-\text{HTSESR}\,(I) * (\text{TESECH}\,(I) - \text{TESROS}\,(I)) \\ &-\text{HTSEFL}\,(I) * (\text{TESECH}\,(I) - \text{TELUCH}\,(I)) \\ &-\text{HTSESL}\,(I) * (\text{TESECH}\,(I) - \text{TESELU}\,(I)) \\ &-\text{HTSENA}\,(I) * (\text{TESECH}\,(I) - \text{TENA}\,(I))] \\ &*\text{DTPLU}/\text{DESECH}\,(I) \end{aligned} \tag{16.4-32}$$

where

$$\text{COSEOA} = \left[(\rho' hu)_{\text{se,i+1/2}} - (\rho' hu)_{\text{se,i-1/2}}\right] \cdot \frac{1}{\Delta z_{\text{i}}} \tag{16.4-33}$$

$$\text{COSEOB} = h^{n}_{\text{se,i}} \cdot \left[(\rho' u)_{\text{se,i+1/2}} - (\rho' u)_{\text{se,i-1/2}}\right] \cdot \frac{1}{\Delta z_{\text{i}}} \tag{16.4-34}$$

HTSEFU $(I) = H'_{\text{se,fu,i}}$ = transfer coefficient between steel and fuel in cell i

HTSECL $(I) = H'_{\text{se,cl,i}}$ = transfer coefficient between steel and cladding in cell i

HTSESR $(I) = H'_{\text{se,sr,i}}$ = transfer coefficient between steel and hexcan wall in cell i

HTSEFL $(I) = H'_{\text{se,fl,i}}$ = transfer coefficient between steel and fuel chunks in cell i

HTSESL $(I) = H'_{\text{se,sl,i}}$ = transfer coefficient between steel and steel chunks in cell i

HTSENA $(I) = H'_{\text{se,mi,i}}$ = transfer coefficient between steel and gas mixture in cell i

The generalized heat-transfer coefficients are defined by:

$$H'_{\text{se,j,i}} = H_{\text{se,j,i}} \cdot \frac{A_{\text{se,j,i}}}{\text{AXMX} \cdot \Delta z_{\text{i}}} \tag{16.4-35}$$

and are described in detail in Section 16.4.3.10.

After the energy change is calculated, the new energy of the steel is calculated according to Eq. (16.4-29), and the temperature of the steel is obtained by using an external function, TESEEG, which uses the enthalpy as argument:

$$T_{\text{se,i}} = T\left(h^{n+1}_{\text{se,i}}\right) \tag{16.4-36}$$

Several checks are performed during the steel temperature calculation in order to avoid numerical difficulties. Thus, if only small amounts of molten steel are present (steel volume less than .1% of the cell volume) the temperature of the steel is set equal to the cladding temperature or, if the steel is surrounded by molten fuel, to the molten fuel temperature.

The condensation/vaporization energy sources/sinks are not included here. They will be introduced later in this chapter, when the energy conservation calculation for steel vapor is presented.

16.4.3.4 Fuel and Steel Chunk Energy Conservation Equations

These equations are solved in the routine LESOEN. Separate equations are solved for the fuel and steel solid chunks, as described below.

16.4.3.4.1 The Energy Conservation Equation for the Fuel Chunks

The energy equation is written in conservative form as follows:

$$\begin{aligned}\frac{\partial}{\partial \text{t}}\left(\rho_{\text{fl,i}} \cdot h_{\text{fl,i}} A_{\text{fl,i}} \cdot \Delta z\right) + \left[(\rho A hu)_{\text{fl,i+1/2}} - (\rho A hu)_{\text{fl,i-1/2}}\right] \\ = Q_{\text{fl,i}} \cdot \rho_{\text{fl,i}} \cdot A_{\text{fl,i}} \cdot \Delta z_{\text{i}} - \sum_{\text{j}} H_{\text{fl,j,i}} \cdot A_{\text{fl,j,i}} \cdot \Delta T_{\text{fl,j,i}}\end{aligned} \tag{16.4-37}$$

where

$Q_{\text{fl,i}}$ = fission power source in the fuel in cell i [J/kg-s]

$A_{\mathrm{fl,i}}$ = area covered by chunks in cell i, when imagined as a continuum with density ρ_{fl}.

Following the same steps as outlined in Eq. (16.4-16) through Eq. (16.4-21), we obtain:

$$h_{fl,\mathrm{i}} = \left\{ -\left[(\rho' hu)_{\mathrm{fl},i+1/2} - (\rho' \mathrm{hu})_{fl,i\text{-}1/2}\right] \cdot \frac{1}{z_{\mathrm{i}}} + h^n_{\mathrm{fl,i}} \cdot \left[(\rho' u)_{\mathrm{fl},i+1/2} - (\rho' u)_{fl,i\text{-}1/2}\right] \cdot \frac{1}{z_{\mathrm{i}}} + Q_{\mathrm{fl,i}} \cdot \rho'_{\mathrm{fl,i}} - \sum_{\mathrm{j}} H'_{\mathrm{fl,j,i}} \cdot T_{\mathrm{fl,j,i}} \right\} \cdot \frac{t}{\rho'^{n+1}_{\mathrm{fl,i}}} \tag{16.4-38}$$

where

$$\begin{aligned} \sum_{\mathrm{j}} H'_{\mathrm{fl,j,i}} \cdot \Delta T_{\mathrm{fl,j,i}} = {} & H'_{\mathrm{fu},fl,\mathrm{i}} \cdot (T_{fl,\mathrm{i}} - T_{\mathrm{fu,i}}) + H'_{\mathrm{se},fl,\mathrm{i}} \cdot (T_{fl,\mathrm{i}} - T_{\mathrm{se,i}}) \\ & + H'_{\mathrm{Na},fl,\mathrm{i}} \cdot (T_{fl,\mathrm{i}} - T_{\mathrm{Na,i}}) + H'_{\mathrm{cl},fl,\mathrm{i}} \cdot (T_{fl,\mathrm{i}} - T_{\mathrm{cl,i}}) \\ & + H'_{\mathrm{sr},fl,\mathrm{i}} \cdot (T_{fl,\mathrm{i}} - T_{\mathrm{sr,i}}) + H'_{\mathrm{ffc},fl,\mathrm{i}} \cdot (T_{fl,\mathrm{i}} - T_{\mathrm{ffc,i}}) \\ & + H'_{\mathrm{ffs},fl,\mathrm{i}} \cdot (T_{fl,\mathrm{i}} - T_{\mathrm{ffs,i}}) + H'_{\mathrm{fl},sl,\mathrm{i}} \cdot (T_{fl,\mathrm{i}} - T_{sl,\mathrm{i}}) \end{aligned} \tag{16.4-39}$$

The energy convective terms in Eq. (16.4-38) are calculated using an upstream differencing approach, similar to that used for Eq. (16.4-23). The convective fluxes are calculated in the routine LEMACO, before the calculation of the new fuel chunk densities, and stored in the array COLVOS (I). The new fuel chunk enthalpy and temperature are then calculated:

$$h^{n+1}_{\mathrm{fl,i}} = h^n_{\mathrm{fl,i}} + \Delta h^n_{\mathrm{fl,i}} \tag{16.4-40}$$

$$T^{n+1}_{fl,\mathrm{i}} = T\left(h^{n+1}_{\mathrm{fl,i}}\right) \tag{16.4-41}$$

16.4.3.4.2 The Energy Conservation for the Steel Chunks

This equation is very similar to the equation used for the fuel chunks, but the fission energy source is not present anymore. Only the final form of the equation is presented here:

$$\Delta h_{sl,\mathrm{i}} = \left\{ -\left[(\rho' hu)_{\mathrm{sl},i+1/2} - (\rho' hu)_{sl,i\text{-}1/2}\right] \cdot \frac{1}{\Delta z_{\mathrm{i}}} + h^n_{sl,\mathrm{i}} \cdot \left[(\rho' u)_{sl,i+1/2} - (\rho' u)_{sl,i\text{-}1/2}\right] \cdot \frac{1}{\Delta z_{\mathrm{i}}} - \sum_{\mathrm{j}} H'_{\mathrm{sl,j,i}} \cdot \Delta T_{sl,\mathrm{j,i}} \right\} \cdot \frac{\Delta t}{\rho'^{n+1}_{sl,\mathrm{i}}} \tag{16.4-42}$$

where

$$\begin{aligned} \sum_{\mathrm{j}} H'_{\mathrm{sl,j,i}} \cdot \Delta T_{sl,\mathrm{j,i}} = {} & H'_{\mathrm{fu},sl,\mathrm{i}} \cdot (T_{sl,\mathrm{i}} - T_{\mathrm{fu,i}}) + H'_{\mathrm{se},sl,\mathrm{i}} \cdot (T_{sl,\mathrm{i}} - T_{\mathrm{se,i}}) \\ & + H'_{\mathrm{Na},sl,\mathrm{i}} \cdot (T_{sl,\mathrm{i}} - T_{\mathrm{Na,i}}) + H'_{\mathrm{cl},sl,\mathrm{i}} \cdot (T_{sl,\mathrm{i}} - T_{\mathrm{cl,i}}) \\ & + H'_{\mathrm{sr},sl,\mathrm{i}} \cdot (T_{sl,\mathrm{i}} - T_{\mathrm{sr,i}}) + H'_{\mathrm{ffc},sl,\mathrm{i}} \cdot (T_{sl,\mathrm{i}} - T_{\mathrm{ffc,i}}) \\ & + H'_{\mathrm{ffs},sl,\mathrm{i}} \cdot (T_{sl,\mathrm{i}} - T_{\mathrm{ffs,i}}) + H'_{\mathrm{fl},sl,\mathrm{i}} \cdot (T_{sl,\mathrm{i}} - T_{fl,\mathrm{i}}) \end{aligned} \tag{16.4-43}$$

The new steel chunk enthalpy and temperature are then calculated as follows:

$$h^{n+1}_{sl,\mathrm{i}} = h^n_{sl,\mathrm{i}} + \Delta h^n_{sl,\mathrm{i}} \tag{16.4-44}$$

$$T^{n+1}_{sl,\mathrm{i}} = T\left(h^{n+1}_{sl,\mathrm{i}}\right) \tag{16.4-45}$$

16.4.3.5 Sodium and Fission-gas Energy Conservation Equation

The sodium and the fission gas are assumed in LEVITATE to be in equilibrium at the same temperature $T_{\text{Na,i}}$. The sodium can exist in the form of superheated vapor or as a two-phase, vapor-liquid mixture. When required by the thermodynamic conditions the two-phase sodium component can become subcooled sodium. The two-phase sodium is assumed to be in thermodynamic equilibrium. The energy conservation equation is solved simultaneously for sodium and fission gas for all cells in the LEVITATE interaction region, i.e., IFMIBT $\leq I \leq$ IFMITP. Because the fission-gas region extends only between IFFIBT $\leq I \leq$ IFFITP, it is possible that some cells contain only sodium, without any fission gas. In these cells, the energy equation for the sodium-fission-gas mixture reduces to a sodium-only equation. Because significant differences exist between the behavior of superheated and two phase sodium, two separate equations are used.

16.4.3.5.1 The Energy Conservation Equation for Superheated Sodium and Fission Gas

The energy equation used for superheated sodium and fission gas is written in conservative form as:

$$\begin{aligned}
&\frac{\partial}{\partial \text{t}}\left(\rho_{\text{Na,i}} h_{\text{Na,i}} + \rho_{\text{fi,i}} h_{\text{fi,i}}\right) \cdot A_{\text{Mi,i}} \cdot \Delta z_{\text{i}} + \left[(\rho A h u)_{\text{Na,i+1/2}}\right. \\
&\quad \left. + (\rho \text{Ahu})_{\text{fi,i+1/2}} - (\rho A h u)_{\text{Na,1-1/2}} - (\rho A h u)_{\text{fi,i-1/2}}\right] \\
&= A_{\text{Mi,i}}\left(\frac{\partial \text{P}_{\text{Na,i}}}{\partial \text{t}} + \frac{\partial \text{P}_{\text{fi,i}}}{\partial \text{t}}\right) \cdot \Delta z_{\text{i}} + A_{\text{Mi,i}} \cdot \left[u_{\text{Mi,i}} \cdot 0.5 \cdot (P_{\text{Na,i}} + P_{\text{fi,i}}\right. \\
&\left. - P_{\text{Na,i-1}} - P_{\text{fi,i-1}}) + U_{\text{Mi,i+1}} \cdot 0.5 \cdot (P_{\text{Na,i+1}} + P_{\text{fi,i+1}} - P_{\text{Na,i}} - P_{\text{fi,i}})\right] \\
&\quad + \sum_{\text{j}} H_{\text{j,Mi,i}} \cdot A_{\text{j,Mi,i}} \cdot \Delta T_{\text{j,Mi,i}}
\end{aligned} \tag{16.4-46}$$

where:

$H_{\text{j,Mi,i}}$ = heat-transfer coefficient between component j and sodium-fission gas mixture in cell i [J/m^2 - s- K]

$A_{\text{j,Mi,i}}$ = heat-transfer area between component j and the gas mixture in cell i [m^2]

$\Delta T_{\text{j,Mi,I}}$ = temperature difference between component j and the gas mixture in cell i [K]

After dividing by AXMX $\cdot \Delta z_{\text{i}}$ and using the definition of generalized densities, we obtain:

$$\begin{aligned}
&\frac{\partial}{\partial \text{t}}\left(\rho'_{\text{Na,i}} h_{\text{Na,i}} + \rho'_{\text{fi,i}} h_{\text{fi,i}}\right) = -\left[(\rho' h u)_{\text{Na,i+1/2}} + (\rho' h u)_{\text{fi,i+1/2}}\right. \\
&\left. - (\rho' h u)_{\text{Na,i-1/2}} - (\rho' h u)_{\text{fi,i-1/2}}\right] \cdot \frac{1}{\Delta z_{\text{i}}} + \theta_{\text{Mi,i}} \cdot \left(\frac{\partial \text{P}_{\text{Na,i}}}{\partial \text{t}} + \frac{\partial \text{P}_{\text{fi,i}}}{\partial \text{t}}\right) \\
&\quad + \theta_{\text{Mi,i}} \cdot \frac{0.5}{\Delta z_{\text{i}}} \cdot \left[u_{\text{Mi,i}} \cdot (P_{\text{Na,i}} + P_{\text{fi,i}} - P_{\text{Na,i-1}} - P_{\text{fi,i-1}})\right. \\
&\left. + u_{\text{Mi,i+1}} \cdot (P_{\text{Na,i+1}} + P_{\text{fi,i+1}} - P_{\text{Na,i}} - P_{\text{fi,i}})\right] + \sum_{\text{j}} H'_{\text{j,Mi,i}} \cdot \Delta T_{\text{j,Mi,i}}
\end{aligned} \tag{16.4-47}$$

where the generalized heat-transfer coefficients are defined by:

$$H'_{\text{j,Mi,i}} = H_{\text{j,Mi,i}} \cdot \frac{A_{\text{j,Mi,i}}}{\text{AXMX} \cdot \Delta z_{\text{i}}} \tag{16.4-48}$$

and the sum of the heat-transfer terms $\sum_{\text{j}} H'_{\text{j,Mi,i}} \cdot \Delta T_{\text{j,Mi,i}}$ is presented in Eq. (16.4-60). Integrating over Δt

and using the identity shown in Eq. (16.4-18) to express the quantity $\Delta(\rho' h)$, we obtain:

$$\begin{aligned}
&\rho'^{n+1}_{\text{Na,i}} \cdot \Delta h_{\text{Na,i}} + \rho'^{n+1}_{\text{fi,i}} \cdot \Delta h_{\text{fi,i}} \\
&= -\left[(\rho' hu)_{\text{Na,i+1/2}} + (\rho' hu)_{\text{fi,i+1/2}}\right. \\
&\quad -h_{\text{Na,i}} \cdot \Delta\rho'_{\text{Na,i}} - h_{\text{fi,i}} \cdot \Delta\rho'_{\text{fi,i}} \\
&\quad \left. -(\rho' hu)_{\text{Na,i-1/2}} - (\rho' hu)_{\text{fi,i-1/2}}\right] \cdot \frac{\Delta t}{\Delta z_{\text{i}}} \\
&+\theta_{\text{Mi,i}} \cdot (\Delta P_{\text{Na,i}} + \Delta P_{\text{fi,i}}) + \theta_{\text{Mi,i}} \cdot \frac{0.5 \cdot \Delta t}{\Delta z_{\text{i}}} \cdot [u_{\text{Mi,i}} \cdot (P_{\text{Na,i}} + P_{\text{fi,i}} \\
&\quad -P_{\text{Na,i-1}} - P_{\text{fi,i-1}}) + u_{\text{Mi,i+1}} \cdot (P_{\text{Na,i+1}} + P_{\text{fi,i+1}} - P_{\text{Na,i}} - P_{\text{fi,i}})] \\
&\quad + \sum_{\text{j}} H'_{\text{j,Mi,i}} \cdot \Delta T_{\text{j,Mi,i}} \cdot \Delta t
\end{aligned} \tag{16.4-49}$$

Two sets of equation relating the thermodynamic characteristics of a gas are used to refine Eq. (16.4-49). These are:

$$h_{\text{Na,i}} = 2.5 R_{\text{Na}} \cdot (T_{\text{Na}} - T_{\text{sat}}) + h_{\text{Na,vap}} \tag{16.4-50}$$

$$h_{\text{fi,i}} = 2.5 R_{\text{fi}} \cdot T_{\text{fi}} \tag{16.4-51}$$

and

$$P_{\text{Na,i}} = R_{\text{Na}} \cdot \rho'_{\text{Na,i}} \cdot T_{\text{Na,i}} \cdot \frac{1}{\theta_{\text{Na,i}}} \tag{16.4-52}$$

$$P_{\text{fi,i}} = R_{\text{fi}} \cdot \rho'_{\text{fi,i}} \cdot T_{\text{fi,i}} \cdot \frac{1}{\theta_{\text{fi,i}}} \tag{16.4-53}$$

where the constant R_{j} is defined as:

$$P_{\text{j}} = \frac{R}{M_{\text{j}}} \tag{16.4-54}$$

with

R = universal gas constant [J/mol - K]

M_{j} = molar mass of gas j [kg/mol]

$T_{\text{sat}} = T_{\text{sat}}(P_{\text{Na,i}})$; $h_{\text{Na,vap}} = h_{\text{Na,vap}}(P_{\text{Na,i}})$.

With $T_{\text{Na,i}} = T_{\text{fi,i}}$ and $\theta_{\text{Na,i}} = \theta_{\text{fi,i}} = \theta_{\text{Mi,i}}$, after expressing the quantity $\Delta T_{\text{j,Mi,i}}$ as:

$$\Delta T_{\text{j,Mi,i}} = T_{\text{j,i}} - T^{n+1}_{\text{Mi,i}} = T_{\text{j,i}} - T_{\text{Mi,i}} - \Delta T_{\text{Mi,i}} \tag{16.4-55}$$

and differencing Eq. (16.4-50) and Eq. (16.4-51) to obtain $\Delta h_{\text{Na,i}}$ and $\Delta h_{\text{fi,i}}$, Eq. (16.4-49) becomes:

$$\begin{aligned}
&\Delta T_{\text{Mi,i}} \left[2.5\rho'^{n+1}_{\text{Na,i}} R_{\text{Na}} + 2.5\rho'^{n+1}_{\text{fi,i}} R_{\text{fi}}\right] + \sum_{\text{j}} H'_{\text{j,Mi,i}} \Delta T_{\text{Mi,i}} \Delta t \\
&= -\left[(\rho' hu)_{\text{Na,i+1/2}} + (\rho' hu)_{\text{fi,i+1/2}} - (\rho' hu)_{\text{Na,i-1/2}} - (\rho' hu)_{\text{fi,i-1/2}}\right] \frac{\Delta t}{\Delta z_{\text{i}}} \\
&\quad -h_{\text{Na,i}} \Delta\rho'_{\text{Na,i}} - h_{\text{fi,i}} \Delta\rho'_{\text{fi,i}} + R_{\text{Na}} \Delta\left(\rho'_{\text{Na,i}} T_{\text{Na,i}}\right) + R_{\text{fi}} \Delta\left(\rho'_{\text{fi,i}} T_{\text{fi,i}}\right) \\
&+\theta_{\text{Mi,i}} \frac{0.5\Delta t}{\Delta z_{\text{i}}} \left[u_{\text{Mi,i}} \left(P_{\text{Na,i}} + P_{\text{fi,i}} - P_{\text{Na,i-1}} - P_{\text{fi,i-1}}\right) + u_{\text{Mi,i+1}} \left(P_{\text{Na,i+1}} + P_{\text{fi,i+1}} - P_{\text{Na,i}} - P_{\text{fi,i}}\right)\right] \\
&\quad + \sum_{\text{j}} H'_{\text{j,Mi,i}} \left(T_{\text{j,i}} - T_{\text{Mi,i}}\right) \Delta t
\end{aligned} \tag{16.4-56}$$

After using the identity:

$$\Delta\left(\rho' T\right) = T^{n}\Delta\rho' + \rho'^{n+1}\cdot\Delta T \tag{16.4-57}$$

and replacing $\Delta\rho'_{\text{Na,i}}$ and $\Delta\rho'_{\text{fi,i}}$ by using the continuity equation, Eq. (16.4-56) becomes:

$$\begin{aligned} T_{\text{Mi,i}}\cdot\left[2.5\rho'^{n+1}_{\text{Na,i}}R_{\text{Na}} + 2.5\rho'^{n+1}_{\text{fi,i}}R_{\text{fi}} + \sum_{\text{j}} H'_{\text{j,Mi,i}}\cdot t - R_{\text{Na}}\rho'^{n+1}_{\text{Na,i}} - R_{\text{fi}}\cdot\rho'^{n+1}_{\text{fi,i}}\right] \\ = -\left[\left(\rho' hu\right)_{\text{Na,i+1/2}} + \left(\rho' hu\right)_{\text{fi,i+1/2}} - \left(\rho' hu\right)_{\text{Na,i-1/2}} - \left(\rho' hu\right)_{\text{fi,i-1/2}}\right]\cdot \\ \frac{t}{z_{\text{i}}} + \left(h_{\text{Na,i}} - T_{\text{Na,i}}\cdot R_{\text{Na}}\right)\cdot\left[\left(\rho' u\right)_{\text{Na,i+1/2}} - \left(\rho' u\right)_{\text{Na,i-1/2}}\right]\frac{t}{z_{\text{i}}} + \left(h_{\text{fi,i}} - T_{\text{fi,i}}R_{\text{fi}}\right) \\ \cdot\left[\left(\rho' u\right)_{\text{fi,i+1/2}} - \left(\rho' u\right)_{\text{fi,i-1/2}}\right]\frac{t}{z_{\text{i}}} + \theta_{\text{Mi,i}}\cdot\frac{0.5\cdot t}{z_{\text{i}}}\cdot\left[u_{\text{Mi,i}}\cdot\left(P_{\text{Na,i}} + P_{\text{fi,i}} - P_{\text{Na,i-1}} - P_{\text{fi,i-1}}\right)\right. \\ \left. + u_{\text{Mi,i+1}}\cdot\left(P_{\text{Na,i+1}} + P_{\text{fi,i+1}} - P_{\text{Na,i}} - P_{\text{fi,i}}\right)\right] + \sum_{\text{j}} H'_{\text{j,Mi,i}}\cdot\left(T_{\text{j,i}} - T_{\text{Mi,i}}\right)\cdot t \end{aligned} \tag{16.4-58}$$

The terms derived from the original $\frac{\partial \text{P}}{\partial \text{t}}$ and, $u\cdot\frac{\partial \text{P}}{\partial \text{z}}$ terms in Eq. (16.4-46) make a negligible contribution to the right hand side of Eq. (16.4-45). The terms are neglected in the initial release version, but have been added, for completeness, in the chunk development version. The left-hand-side coefficient is replaced by:

$$\begin{aligned} \text{AUXLR} = 2.5\rho'^{n+1}_{\text{Na,i}}R_{\text{Na}} + 2.5\rho'^{n+1}_{\text{fi,i}}\cdot R_{\text{fi}}\cdot\sum_{\text{j}} H'_{\text{j,mi,i}}\Delta t \\ - R_{\text{Na}}\rho'^{n+1}_{\text{Na,i}} - R_{\text{fi}}\rho'^{n+1}_{\text{fi,i}} \end{aligned} \tag{16.4-59}$$

After dividing Eq. (16.4-58) by AUXLR, the new mixture temperature is calculated as follows:

$$\begin{aligned} \text{TENA}\left(I\right) = \text{TENA}\left(I\right) + \left(-\text{COENCH} + \text{COHELP} + \text{DTPLU}\right) \\ *\left(\text{HTNACL}\left(I\right) * \left(\text{TECLOL}\left(I\right) - \text{TENA}\left(I\right)\right) + \text{HTSENA}\left(I\right) * \left(\text{TESEOL}\left(I\right) - \text{TENA}\left(I\right)\right)\right) \\ + \text{HTNASR}\left(I\right) * \left(\text{TESROL}\left(I\right) - \text{TENA}\left(I\right)\right) + \text{HTFUNA}\left(I\right) * \left(\text{TEFUOL}\left(I\right) = \text{TENA}\left(I\right)\right) \\ + \text{HTNAFL}\left(I\right) * \left(\text{TEFLOL}\left(I\right) - \text{TENA}\left(I\right)\right) + \text{HTNASL}\left(I\right) * \left(\text{TESLOL}\left(I\right) = \text{TENA}\left(I\right)\right) \\ + \text{HTFCNA} * \left(\text{TEFFCO}\left(I\right) - \text{TENA}\left(I\right)\right) + \text{HTFSNA} * \left(\text{TEFFSO}\left(I\right) - \text{TENA}\left(I\right)\right) / \text{AUXLR} \end{aligned} \tag{16.4-60}$$

where:

$$\begin{aligned} \text{COENCH} = \left[\left(\rho' hu\right)_{\text{Na,i+1/2}} + \left(\rho' hu\right)_{\text{fi,i+1/2}} - \left(\rho' hu\right)_{\text{Na,i-1/2}}\right. \\ \left. - \left(\rho' hu\right)_{\text{fi,i-1/2}}\right]\cdot\frac{\Delta t}{\Delta z_{\text{i}}} \end{aligned} \tag{16.4-61}$$

$$\begin{aligned} \text{COHELP} = h_{\text{Na,i}}\cdot\left[\left(\rho' u\right)_{\text{Na,i+1/2}} - \left(\rho' u\right)_{\text{Na,i-1/2}}\right]\cdot\frac{\Delta t}{\Delta z_{\text{i}}} \\ + h_{\text{fi,i}}\cdot\left[\left(\rho' u\right)_{\text{fi,i+1/2}} - \left(\rho' u\right)_{\text{fi,i-1/2}}\right]\cdot\frac{\Delta t}{\Delta z_{\text{i}}} \end{aligned} \tag{16.4-62}$$

$\text{HTNACL}\left(I\right) = H'_{\text{cl,Na,i}}$ = transfer coefficient between cladding and mixture in cell i

$\text{HTSENA}\left(I\right) = H'_{\text{se,Na,i}}$ = transfer coefficient between steel and gas mixture in cell i

$\text{HTNASR}\left(I\right) = H'_{\text{sr,Na,i}}$ = transfer coefficient between hexcan wall and gas mixture in cell i

$\text{HTFUNA}\left(I\right) = H'_{\text{fu,Na,i}}$ = transfer coefficient between molten fuel and gas mixture in cell i

$\text{HTNAFL}\left(I\right) = H'_{\text{fl,Na,i}}$ = transfer coefficient between steel chunks and mixture in cell i

$\text{HTFCNA} = H'_{\text{ffc,Na,i}}$ = transfer coefficient between frozen fuel on cladding and gas mixture in cell i

HTFSNA $= H'_{\text{ffs,Na,i}}$ = transfer coefficient between frozen fuel on can wall and gas mixture in cell i

The convective enthalpy fluxes used in Eq. (16.4-59) are calculated using an upstream differencing approach:

If $u'_{\text{Na,i}} \geq 0$

$$(\rho' hu)_{\text{Na,i-1/2}} = \rho'_{\text{Na,i-1}} \cdot h_{\text{Na,i-1}} \cdot u'_{\text{Na,i}} \tag{16.4-63}$$

If $u'_{\text{Na,i}} < 0$

$$(\rho' hu)_{\text{Na,i-1/2}} = \rho'_{\text{Na,i}} \cdot h_{\text{Na,i}} \cdot u'_{\text{Na,i}} \tag{16.4-64}$$

The enthalpy $h_{\text{Na,i}}$ is calculated as follows

If $X_{\text{Na,i}} \geq 1$ (single-phase sodium vapor),

$$h_{\text{Na,i}} = h_{\text{Na,vap}}\left(P_{\text{Na,i}}\right) + 2.5 \cdot R_{\text{Na}} \cdot \left(T_{\text{Na,i}} - T_{\text{sat}}\left(P_{\text{Na,i}}\right)\right) \tag{16.4-65}$$

If $0 \leq X < 1$ (two-phase sodium), then

$$h_{\text{Na,i}} = h_{\text{Na,liq}}\left(T_{\text{Na,i}}\right) + h_{\text{Na,fg}}\left(T_{\text{Na,i}}\right) \cdot X_{\text{Na,i}} \tag{16.4-66}$$

where

$h_{\text{Na,fg}}\left(T_{\text{Na,i}}\right)$ = the heat of vaporization of sodium at the temperature $T_{\text{Na,i}}$

$h_{\text{Na,liq}}\left(T_{\text{Na,i}}\right)$ = enthalpy of liquid sodium on the saturation curve at $T_{\text{Na,i}}$

$h_{\text{Na,vap}}\left(P_{\text{Na,i}}\right)$ = enthalpy of sodium vapor on the saturation curve at pressure $P_{\text{Na,i}}$

If injection of fission gas has taken place in the previous time step, a correction of the temperature is made to account for the enthalpy of the injected gas. When the injection is calculated, this gas is assumed to be at the same temperature as the sodium-fission gas mixture in the channel in order to avoid recalculating the mixture temperature.

$$T_{\text{na,i}} = T_{\text{Na,i}} + \left(T_{\text{fi,ejected,i}} - T_{\text{Na,i}}\right) \cdot \Delta\rho'_{\text{fi,ejected,i}} \cdot 2.5 \frac{R_{\text{fi}}}{\text{AUXLR}} \tag{16.4-67}$$

The pressure of the mixture is then calculated as:

$$P_{\text{ch,i}} = P_{\text{Na,i}} + P_{\text{fi,i}} \tag{16.4-68}$$

where $P_{\text{Na,i}}$ and $P_{\text{fi,i}}$ are calculated using Eq. (16.4-52) and Eq. (16.4-53). In using this equation for sodium, it is recognized that in the neighborhood of the saturation curve the behavior of the sodium vapor will deviate from the perfect-gas behavior. Thus, the constant $R_{\text{Na,i}}$ is allowed to vary so that the pressure calculated by Eq. (16.4-52) will match the saturation pressure when the mixture is right on the saturation curve. The parameter used to decide if this procedure is necessary is the ratio:

$$r_{\text{i}} = \frac{\rho_{\text{Na,i}}}{\rho_{\text{Na,vap}}\left(T_{\text{Na,i}}\right)} \tag{16.4-69}$$

where

$\rho_{\text{Na,i}}$ = physical density of the sodium vapor

$\rho_{\text{Na,vap}}$ = physical density of the saturated sodium vapor

If $r_{\text{i}} < r_{\text{min}}$ the sodium vapor is far enough from saturation and the perfect-gas law is satisfactory. Otherwise, the constant $R'_{\text{Na,i}}$ is defined as:

$$R'_{\text{Na,i}} = \frac{\left[R_{\text{Na,i}} \cdot (1 - r) + R_{\text{Na,sat,i}} \cdot (r - r_{\text{min}})\right]}{(1 - r_{\text{min}})} \tag{16.4-70}$$

where

$$R_{\text{Na,sat,i}} = \frac{P_{\text{Na,sat}}\left(T_{\text{Na,i}}\right) \cdot \theta_{\text{Na,i}}}{\rho'_{\text{Na,i}} \cdot T_{\text{Na,i}}} \tag{16.4-71}$$

and r_{min} is the ratio obtained from the sodium thermodynamic properties tables for a wide range of interest; currently $r_{\text{min}} = .067$.

This derivation assumes that the sodium is in the superheated vapor region for the entire duration of the time step Δt. Occasionally, however, the sodium vapor might become saturated and this assumption would no longer hold. In the two-phase region the temperature changes are smaller than those in the single-phase region, since much of the energy lost will lead to condensation, rather than to a change in temperature. Thus, where necessary, a correction is performed which reduces the temperature drop in the two-phase region by bringing the final sodium temperature close to the saturation curve. The correction is considered necessary whenever the following condition is satisfied:

$$P_{\text{Na,i}} - P_{\text{Na,sat,i}} > \Delta P_{\text{constant}} \tag{16.4-72}$$

where $\Delta P_{\text{constant}}$ is a built-in constant, currently set to $0.05 \cdot 10^5$ Pa. This procedure is illustrated in Figure 16.4.13. The corrected sodium temperature is calculated as follows:

$$T_{\text{Na,i}} = \frac{\left(T_1 \cdot \Delta P_{\text{o}} - T_{\text{o}} \cdot \Delta P_1\right)}{\left(\Delta P_{\text{o}} - \Delta P_1\right)} \tag{16.4-73}$$

where

T_{o} = sodium temperature at the beginning of the time step

T_1 = calculated new sodium temperature, from Eq. (16.4-67)

$\Delta P_{\text{o}}, \Delta P_1 = P_{\text{Na,sat,i}} - P_{\text{Na,i}}$ at the beginning and end of time step, respectively.

16.4.3.5.2 The Energy Equation for Two-phase Sodium and Fission Gas

The same considerations as before are made to arrive at the Eq. (16.4-49), but the compressible term, $\theta \frac{\partial \text{P}}{\partial \text{t}}$, is not included for the two-phase component. To express the quantity $\Delta h_{\text{Na,i}}$, we use the identity:

$$h_{\text{Na,i}} = h_{\text{Na,liq,i}} \cdot \left(1 - X_{\text{Na,i}}\right) + h_{\text{Na,vap,i}} \cdot X_{\text{Na,i}} \tag{16.4-74}$$

Thus:

$$\begin{aligned} \Delta h_{\text{Na,i}} = \left(1 - X_{\text{Na,i}}\right) \cdot \frac{\text{dh}_{\text{liq}}}{\text{dT}} \cdot \Delta T_{\text{Na,i}} + X_{\text{Na,i}} \cdot \frac{\text{dh}_{\text{vap}}}{\text{dT}} \cdot \Delta T_{\text{Na,i}} \\ + \left(h_{\text{vap,i}} - h_{\text{liq,i}}\right) \cdot \Delta X_{\text{Na,i}} \end{aligned} \tag{16.4-75}$$

Using the definitions:

$$C_{\text{liq}} = \frac{\text{dh}_{\text{liq}}}{\text{dT}} \tag{16.4-76}$$

$$C_{\text{vap}} = \frac{\text{dh}_{\text{vap}}}{\text{dT}} \tag{16.4-77}$$

$$X = \frac{v - v_{\text{liq}}}{v_{\text{vap}} - v_{\text{liq}}} \tag{16.4-78}$$

$$h_{\text{lv,i}} = h_{\text{vap,i}} - h_{\text{liq,i}} \tag{16.4-79}$$

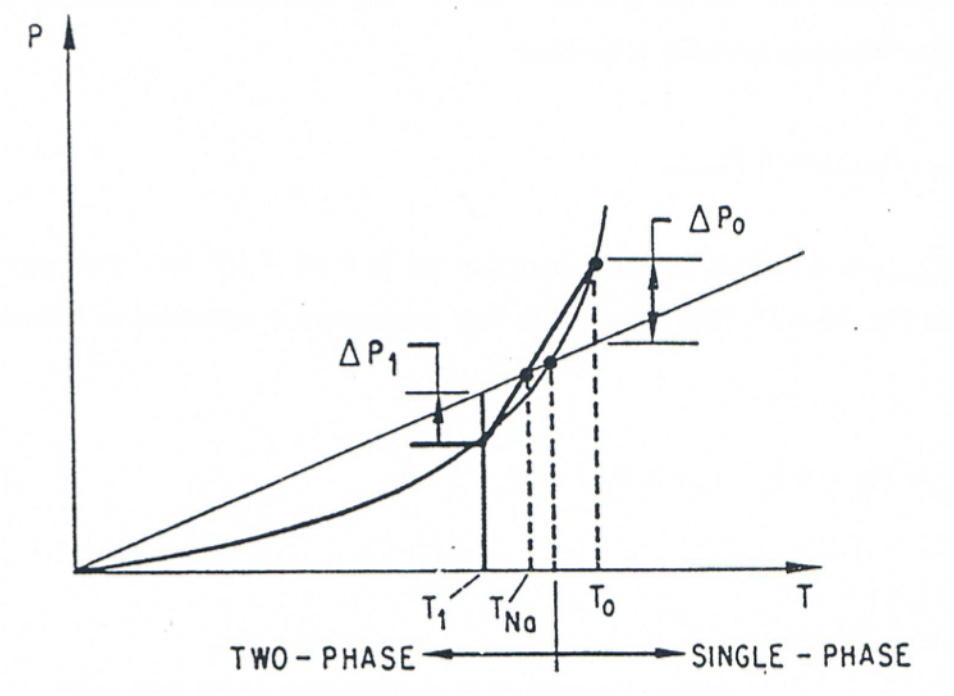


Figure 16.4.13: Correction for Sodium Vapor Transition to a Two-phase Mixture

Eq. (16.4-74) becomes:

$$\begin{aligned}\Delta h_{\text{Na,i}} &= (1 - X_{\text{Na,i}})\, C_{\text{liq}} \cdot \Delta T_{\text{Na,i}} + X_{\text{Na,i}} C_{\text{vap}} \cdot \Delta T_{\text{Na,i}} + h_{\text{lv,i}} \frac{\partial \text{X}}{\partial \text{t}} \cdot \Delta t \\ &= (1 - X_{\text{Na},i}) \;\cdot\; C_{\text{liq}} \;\cdot\; \Delta T_{\text{Na},i} + X_{\text{Na},i} C_{\text{vap}} \;\cdot\; \Delta T_{\text{Na},i} \\ &+ h_{\text{lv},i} \cdot \frac{1}{v_{\text{vap}} - v_{\text{liq}}} \frac{\partial v}{\partial t} \cdot \Delta t - h_{\text{lv},i} \frac{v - v_{\text{liq}}}{(v_{\text{vap}} - v_{\text{liq}})^2} \cdot \frac{dv_v}{\text{dT}} \Delta T_{\text{Na},i}\end{aligned} \tag{16.4-80}$$

By substituting:

$$v = \frac{1}{\rho} = \frac{1}{\left(\frac{\rho'}{\theta}\right)} \tag{16.4-81}$$

in Eq. (16.4-80), we obtain:

$$\begin{aligned}\Delta h_{\text{Na,i}} &= (1 - X_{\text{Na,i}}) \cdot C_{\text{liq}} \cdot \Delta T_{\text{Na,i}} + X_{\text{Na,i}} \cdot C_{\text{vap}} \cdot \Delta T_{\text{Na,i}} \\ &+ h_{\text{lv,i}} \cdot \frac{1}{v_{\text{vap}} - v_{\text{liq}}} \cdot \frac{\theta^2}{\rho'^2} \left[\frac{1}{\theta} \cdot \frac{\partial \rho'}{\partial \text{t}} \cdot \Delta t - \frac{\partial \theta}{\partial \text{t}} \cdot \Delta t\right] \\ &+ h_{\text{lv,i}} \cdot X_{\text{Na,i}} \cdot \frac{1}{v_{\text{vap}} - v_{\text{liq}}} \cdot \frac{\partial v_{\text{v}}}{\partial \text{T}} \cdot \Delta T\end{aligned} \tag{16.4-82}$$

Finally, substituting Eq. (16.4-82) into Eq. (16.4-49) and keeping the terms containing ΔT on the left-hand side, we have:

$$T_{\text{Mi,i}}\left\{\rho'^{n+1}_{\text{Na,i}}\cdot\left[(1-X_{\text{na,i}})\cdot C_{\text{liq}}+X_{\text{Na,i}}\cdot C_{\text{vap}}+h_{\text{lv,i}}\cdot\frac{1}{\nu_{\text{vap}}-\nu_{\text{liq}}}\cdot\frac{\partial\nu_{\text{v}}}{\partial \text{T}}\cdot X_{\text{Na,i}}\right]+2.5\rho'^{n+1}_{\text{fi,i}}R_{\text{fi}}\right\}=$$
$$-\left[(\rho' hu)_{\text{Na,i+1/2}}+(\rho' hu)_{\text{fi,i+1/2}}-(\rho' hu)_{\text{Na,i-1/2}}-(\rho' hu)_{\text{fi,i-1/2}}\right]\cdot\frac{t}{z_{\text{i}}}$$
$$+h_{\text{lv,i}}\cdot\frac{1}{\nu_{\text{vap}}-\nu_{\text{liq}}}\left[\frac{\theta^{n+1}_{\text{Mi}}}{\rho'^{n+1}_{\text{Na,i}}}\cdot\rho'_{\text{Na,i}}-\theta_{\text{Mi}}\right]$$
$$-h_{\text{Na,i}}\cdot\rho'_{\text{Na,i}}-h_{\text{fi}}\cdot\rho'_{\text{fi,i}}+\sum_{\text{j}}H'_{\text{j,Mi,i}}\cdot T_{\text{j,Mi,i}}\cdot t \tag{16.4-83}$$

We now rewrite:

$$\Delta T_{\text{j,Mi,i}}=T_{\text{j,i}}-(T_{\text{Mi,i}}+\Delta T_{\text{Mi,i}}) \tag{16.4-84}$$

$$\Delta T_{\text{j,Mi,i}}=T_{\text{j,i}}-(T_{\text{Mi,i}}+\Delta T_{\text{Mi,i}}) \tag{16.4-85}$$

$$\Delta\rho'_{\text{fi,i}}=-\left[(\rho u)_{\text{fi,i+1/2}}-(\rho u)_{\text{fi,i-1/2}}\right]\cdot\frac{\Delta t}{\Delta z_{\text{i}}} \tag{16.4-86}$$

and, after some regrouping following the same procedure as in the case in the single-phase sodium, we obtain:

$$T^{n+1}_{\text{Mi,i}}=T^{n}_{\text{Mi,i}}+\left\{-\text{COENCH}+h_{\text{lv,i}}\cdot\frac{1}{\nu_{\text{vap}}-\nu_{\text{liq}}}\cdot\frac{\theta^{n+1}_{\text{Mi,i}}}{\rho'^{n+1}_{\text{Mi,i}}}\cdot\rho'_{\text{Na,i}}-\theta_{\text{Mi,i}}+\text{COHELP}\right.$$
$$+T\cdot[H'_{\text{cl,Mi,i}}\cdot(T_{\text{cl,OS,i}}-T_{\text{Mi,i}})+H'_{\text{sr,Mi,i}}(T_{\text{sr,OS,i}}-T_{\text{Mi,i}})+H'_{\text{fu,Mi,i}}(T_{\text{fu,i}}-T_{\text{Mi,i}})$$
$$+H'_{\text{se,Mi,i}}(T_{\text{se,i}}-T_{\text{Mi,i}})+H'_{fl,\text{Mi,i}}(T_{fl,\text{i}}-T_{\text{Mi,i}})+H'_{sl,\text{Mi,i}}(T_{sl,\text{i}}-T_{\text{Mi,i}})$$
$$\left.+H'_{\text{ffc,Mi,i}}(T_{\text{ffc,i}}-T_{\text{Mi,i}})+H'_{\text{ffs,Mi,i}}(T_{\text{ffs,i}}-T_{\text{Mi,i}})]\right\}/\text{AUXLR} \tag{16.4-87}$$

where

$$\text{COENCH}=\left[(\rho' hu)_{\text{Na,i+1/2}}+(\rho' hu)_{\text{fi,i+1/2}}\right.$$
$$\left.-(\rho' hu)_{\text{Na,i-1/2}}-(\rho' hu)_{\text{fi,i-1/2}}\right]\cdot\frac{\Delta t}{\Delta z_{\text{i}}} \tag{16.4-88}$$

$$\text{COHELP}=h_{\text{Na,i}}\cdot\left[(\rho' u)_{\text{Na,i+1/2}}-(\rho' u)_{\text{Na,i-1/2}}\right]\cdot\frac{\Delta t}{\Delta z_{\text{i}}}$$
$$+h_{\text{fi,i}}\cdot\left[(\rho' u)_{\text{fi,i+1/2}}-(\rho' u)_{\text{fi,i-1/2}}\right]\cdot\frac{\Delta t}{\Delta z_{\text{i}}} \tag{16.4-89}$$

$$\text{AUXLR}=\rho'^{n+1}_{\text{Na,i}}\cdot\left[(1-X_{\text{Na,i}})\cdot C_{\text{liq}}+X_{\text{Na,i}}\cdot C_{\text{vap}}-h_{\text{vi,i}}\frac{1}{v_{\text{vap}}-v_{\text{liq}}}\right.$$
$$\left.\cdot\frac{\text{dv}_{\text{v}}}{\text{dT}}\cdot X_{\text{Na,i}}\right]+2.5\rho'^{n+1}_{\text{fi,i}}R_{\text{fi}}+\sum_{\text{j}}H'_{\text{j,Mi,i}}\cdot\Delta t \tag{16.4-90}$$

The heat-transfer coefficients are the same as those used in the single-phase energy equation. The pressure of the mixture is then calculated as follows:

$$P_{\text{ch,i}}=P_{\text{Na,i}}+P_{\text{fi,i}} \tag{16.4-91}$$

where

$$P_{\text{Na,i}}=P_{\text{sat}}(T_{\text{Na,i}}) \tag{16.4-92}$$

$$P_{\text{fi,i}} = R_{\text{fi}} \cdot \rho'_{\text{fi,i}} \cdot \frac{T_{\text{fi,i}}}{\theta_{\text{Mi,i}}} \tag{16.4-93}$$

However, if significant amounts of liquid sodium are present (more than 30% of the volume fraction), the compressibility of the liquid sodium is taken into account in calculating the partial fission-gas pressure. Eq. (16.4-93) is written in the form:

$$P_{\text{fi,i}} = \frac{R_{\text{fi}} \cdot \rho'_{\text{fi,i}} \cdot T_{\text{fi,i}}}{\theta_{\text{Mi,i}} + \Delta\theta_{\text{Na,liq}}} = \frac{R_{\text{fi}} \cdot \rho'_{\text{fi,i}} \cdot T_{\text{fi,i}}}{\theta_{\text{Mi,i}} + \theta_{\text{Na,liq}} \cdot P_{\text{fi,i}} \cdot C_{\Delta P,\text{Na}}} \tag{16.4-94}$$

where

$$C_{\Delta\text{P,Na}} = \frac{\left(\frac{\Delta V}{V}\right)_{\text{Na,liq}}}{\Delta P} \tag{16.4-95}$$

The value of $C_{\Delta\text{P,Na}}$ is given by the input constant CMNL. By solving Eq. (16.4-94) and retaining only the positive root, we obtain:

$$P_{\text{fi,i}} = \frac{-\theta_{\text{Mi,i}} + \sqrt{\theta^2_{\text{Mi,i}} + 4\theta_{\text{Na,liq}} + C_{\Delta\text{P,Na}} \cdot R_{\text{fi}} \cdot \rho'_{\text{fi,i}} \cdot T_{\text{fi,i}}}}{2 \cdot \theta_{\text{Na,liq}} \cdot C_{\Delta\text{P,Na}}} \tag{16.4-96}$$

16.4.3.6 Fuel Vapor Energy Conservation Equation

The fuel vapor energy conservation equation is solved in the routine LEFUVA, which also models the fuel vaporization/condensation processes. These processes are not included in the formulation of the liquid fuel energy conservation equation, which is solved in the LESOEN routine. Thus, the mass and temperature of the liquid fuel is corrected in the routine LEFUVA, when necessary, to account for the mass and energy sources/sinks due to condensation and vaporization. Because the fuel vapor effects become dominant only during high-power transients that are associated with very short time periods, the fuel vapor is assumed *not* to be in equilibrium with the liquid fuel. The treatment of the vaporization and condensation processes will be described in detail later in this chapter.

16.4.3.6.1 Energy Conservation for Superheated Fuel Vapor

First, the energy equation is solved by assuming that all fuel vapor is initially superheated. This assumption is consistent with the method of solution, as all condensation and vaporization events are calculated in the routine LEFUVA, and the fuel vapor remaining in each cell at the end of this routine is always superheated, or, in the limit, saturated dry. Condensation or vaporization effects that might occur during the current time step are ignored during this first step. The energy equation is written in conservative form:

$$\begin{aligned}\frac{\partial}{\partial \text{t}} \left(\rho_{\text{fv,i}} h_{\text{fv,i}} \cdot A_{\text{Mi,i}} \cdot \Delta z_{\text{i}}\right) + \left[(\rho A h u)_{\text{fv,i+1/2}} - (\rho A h u)_{\text{fv,i-1/2}}\right] \\ = A_{\text{Mi,i}} \frac{\partial \text{P}_{\text{fv,i}}}{\partial \text{t}} \cdot \Delta z_{\text{i}} + A_{\text{Mi,i}} \cdot 0.5 \cdot [u_{\text{Mi,i}} \cdot (P_{\text{fv,i}} - P_{\text{fv,i-1}}) \\ + u_{\text{Mi,i+1}} \cdot (P_{\text{fv,i+1}} - P_{\text{fv,i}})] + \sum_{\text{j}} H_{\text{j,fv,i}} A_{\text{j,fv,i}} \cdot \Delta T_{\text{j,fv,i}} \\ + Q_{\text{fu,i}} \cdot \rho_{\text{fu,i}} \cdot A_{\text{Mi,i}} \cdot \Delta z_{\text{i}}\end{aligned} \tag{16.4-97}$$

After dividing by AXMX $\cdot \Delta z_{\mathrm{i}}$ and using the definition of generalized densities, we obtain:

$$\begin{aligned}\frac{\partial}{\partial \mathrm{t}}\left(\rho'_{\mathrm{fv,i}} h_{\mathrm{fv,i}}\right) = & -\left[(\rho' hu)_{\mathrm{fv,i+1/2}} - (\rho' hu)_{\mathrm{fv,i\text{-}1/2}}\right] \cdot \frac{1}{\Delta z_{\mathrm{i}}} \\ & + \theta_{\mathrm{Mi,i}} \cdot \frac{\partial \mathrm{P}_{\mathrm{fv,i}}}{\partial \mathrm{t}} + \sum_{\mathrm{j}} H'_{\mathrm{j,fv,i}} \cdot \Delta T_{\mathrm{j,fv,i}} + Q_{\mathrm{fu,i}} \cdot \rho'_{\mathrm{fv}} \\ & + \theta_{\mathrm{Mi,i}} \cdot \frac{0.5}{\Delta z_{\mathrm{i}}} \left[U_{\mathrm{Mi,i}} \cdot (P_{\mathrm{fv,i}} - P_{\mathrm{fv,i\text{-}1}}) + U_{\mathrm{Mi,i+1}} \cdot (P_{\mathrm{fv,i+1}} - P_{\mathrm{fv,i}})\right]\end{aligned} \tag{16.4-98}$$

Integrating over Δt and using the identity in Eq. (16.4-18) to express the quantity $\Delta\left(\rho' h\right)$, we obtain:

$$\begin{aligned}\rho'^{n+1}_{\mathrm{fv,i}} \cdot \Delta h_{\mathrm{fv,i}} = & -\left[(\rho' hu)_{\mathrm{fv,i+1/2}} - (\rho' hu)_{\mathrm{fv,i\text{-}1/2}}\right] \cdot \frac{\Delta t}{\Delta z_{\mathrm{i}}} - h_{\mathrm{fv,i}} \cdot \Delta \rho'_{\mathrm{fv,i}} \\ & + \theta_{\mathrm{Mi,i}} \cdot \Delta P_{\mathrm{fv,i}} + \sum_{\mathrm{j}} H'_{\mathrm{j,fv,i}} \cdot \Delta T_{\mathrm{j,fv,i}} \cdot \Delta t + Q_{\mathrm{fu,i}} \cdot \rho'_{\mathrm{fv}} \cdot \Delta t \\ & + \theta_{\mathrm{Mi,i}} \cdot \frac{0.5 \cdot \Delta t}{\Delta z_{\mathrm{i}}} \cdot \left[u_{\mathrm{Mi,i}} \cdot (P_{\mathrm{fv,i}} - P_{\mathrm{fv,i\text{-}1}}) \right. \\ & \left. + u_{\mathrm{Mi,i+1}} \cdot (P_{\mathrm{fv,i+1}} - P_{\mathrm{fv,i}})\right]\end{aligned} \tag{16.4-99}$$

The fuel vapor is only present in calculations for short periods of time during high overpower transients. In addition, the gas-gas heat transfer is generally much more efficient than the gas-liquid or gas-surface heat transfer. For these reasons the only heat-transfer term maintained in Eq. (16.4-97) is the heat transfer between the fuel vapor and the sodium-fission gas mixture. Furthermore, because of the lack of experimental data on the gas-gas heat transfer, this term has been formulated such that it represents a sizable fraction of the transferable energy, i.e., the temperatures of the fuel vapor and gas mixture will equilibrate quite fast, within 10-50 time steps. The enthalpy of the fuel vapor at a certain temperature is:

$$h_{\mathrm{fv,i}} = h_{\mathrm{fv,sat}}\left(P_{\mathrm{fv,i}}\right) + 2.5 \cdot R_{\mathrm{fv}} \cdot \left[T_{\mathrm{fv,i}} - T_{\mathrm{fv,sat,i}}\left(P_{\mathrm{fv,i}}\right)\right] \tag{16.4-100}$$

A measure of the transferable enthalpy is obtained by assuming the lowest final temperature to be the temperature of the sodium-fission gas mixture. Thus:

$$\Delta h_{\mathrm{fv,i}} = 2.5 \cdot R_{\mathrm{fv}} \left[T_{\mathrm{fv,i}} - T^{n+1}_{\mathrm{Na,i}}\right] \tag{16.4-101}$$

The total transferable enthalpy in cell i, during the time Δt is:

$$\Delta h_{\mathrm{fv,i}} \cdot \rho_{\mathrm{fv,i}} \cdot A_{\mathrm{Mi,i}} \cdot \Delta z_{\mathrm{i}} = 2.5 \cdot R_{\mathrm{fv}} \cdot \left[T_{\mathrm{fv,i}} - T^{n+1}_{\mathrm{Na,i}}\right] \cdot \rho_{\mathrm{fv,i}} \cdot A_{\mathrm{Mi,i}} \cdot \Delta z_{\mathrm{i}} \tag{16.4-102}$$

The fraction transferred is CFHTAX, which is defined as follows:

When $\Delta t \cdot 10^3 \leq 0.1$

$$\mathrm{CFHTAX} = \Delta t \cdot 10^3 \tag{16.4-103}$$

When $\Delta t \cdot 10^3 > 0.1$

$$\mathrm{CFHTAX} = 0.1 \tag{16.4-104}$$

After multiplying by CFHTAX and dividing by AXMX $\cdot \Delta z_{\mathrm{i}}$, which was done for the original equation, the heat-transfer term in Eq. (16.4-99) is replaced by:

$$\sum_{\mathrm{j}} H'_{\mathrm{j,f,i}} \cdot \Delta T_{\mathrm{j,fv,i}} \cdot \Delta t \rightarrow -2.5 \cdot R_{\mathrm{fv,i}} \cdot \mathrm{CFHTAX} \cdot \left[T_{\mathrm{fv,i}} - T^{n+1}_{\mathrm{Na,i}}\right] \cdot \rho'_{\mathrm{fv,i}} \tag{16.4-105}$$

Substituting Eq. (16.4-105) in Eq. (16.4-99) and rewriting the $\theta_{\text{Mi,i}} \cdot \Delta P_{\text{fv,i}}$ and $\rho_{\text{fv,i}}^{n+1}$ terms in a manner similar to that used for the sodium-gas mixture, in Eq. (16.4-50) through Eq. (16.4-56), we obtain:

$$\begin{aligned}\Delta T_{\text{fv,i}} \left[2.5\left(\rho'\right)_{\text{fv,i}}^{n+1} R_{\text{fv}} - \left(\rho'\right)_{\text{fv,i}}^{n+1} R_{\text{fv}}\right] = \\ -\left[\left(\rho' hu\right)_{\text{fv,i+1/2}} - \left(\rho' hu\right)_{\text{fv,i-1/2}}\right] \frac{\Delta t}{\Delta z_{\text{i}}} \\ +\left(h_{\text{fv,i}} - T_{\text{fv,i}} R_{\text{fv}}\right)\left[\left(\rho' u\right)_{\text{fv,i+1/2}} - \left(\rho' u\right)_{\text{fv,i-1/2}}\right] \frac{\Delta t}{\Delta z_{\text{i}}} \\ +\theta_{\text{Mi,i}} \frac{0.5\Delta t}{\Delta z_{\text{i}}}\left[u_{\text{Mi,i}}\left(P_{\text{fv,i}} - P_{\text{fv,i-1}}\right) + u_{\text{Mi,i+1}}\left(P_{\text{fv,i+1}} - P_{\text{fv,i}}\right)\right] \\ -2.5 R_{\text{fv}} \text{CHFTAX}\left(T_{\text{fv,i}} - T_{\text{Na,i}}^{n+1}\right) \rho'_{\text{fv,i}} + Q_{\text{fu}} \rho'_{\text{fv}} \Delta t\end{aligned} \tag{16.4-106}$$

The terms derived from the original $\frac{\partial P}{\partial t}$ and $\frac{\partial P}{\partial z}$ terms in Eq. (16.4-97) make a negligible contribution to the right hand side of Eq. (16.4-106). These terms were neglected in the initial release version, but have been added for completeness, in the chunk development version. The left-hand-side coefficient is replaced by:

$$\text{AUXLR} = 2.5 \cdot \rho'^{n+1}_{\text{fv,i}} \cdot R_{\text{fv}} - \rho'^{n+1}_{\text{fv,i}} \cdot R_{\text{fv}} \tag{16.4-107}$$

The terms in Eq. (16.4-107) have not been combined because the term $\rho'^{n+1}_{\text{fv,i}} \cdot R_{\text{fv}}$ originates from the $\frac{\partial P}{\partial t}$ term and was dropped in the initial release version, together with the terms originating from the u $\frac{\partial P}{\partial u}$ term. The new fuel vapor temperature is calculated as follows:

$$\begin{aligned}\text{TEFUVA}\left(I\right) = \text{TEFUVA}\left(I\right) + \left(-\text{COENCH} + \text{COHELP} + \text{HSFU}\left(I\right) * \text{DEFVCH}\left(I\right) * \right. \\ \left. \text{DTPL} - \text{DEFVCH}\left(I\right) * \text{CFHTFN} * \left(\text{TEFUVA}\left(I\right) - \text{TENA}\left(I\right)\right)\right) / \text{AUXLR}\end{aligned} \tag{16.4-108}$$

where

$$\text{COENCH} = \left[\left(\rho' hu\right)_{\text{fv,i+1/2}} - \left(\rho' hu\right)_{\text{fv,i-1/2}}\right] \cdot \frac{\Delta t}{\Delta z_{\text{i}}} \tag{16.4-109}$$

$$\text{COHELP} = h_{\text{fv,i}} \cdot \left[\left(\rho' u\right)_{\text{fv,i+1/2}} - \left(\rho' u\right)_{\text{fv,i-1/2}}\right] \cdot \frac{\Delta t}{\Delta z_{\text{i}}} \tag{16.4-110}$$

and

$$\text{DEFVCH}\left(I\right) = \rho'_{\text{fv,i}} \tag{16.4-111}$$

$$\text{CFHTFN} = 2.5 \cdot R_{\text{fv}} \cdot \text{CFHTAX} \tag{16.4-112}$$

The new fuel vapor pressure is then calculated using the new temperatures:

$$P_{\text{fv,i}}^{n+1} = R_{\text{fv}} \cdot \rho'^{n+1}_{\text{fv,i}} \cdot T_{\text{fv,i}}^{n+1} / \theta_{\text{Mi,i}}^{n+1} \tag{16.4-113}$$

16.4.3.6.2 Condensation of the Fuel Vapor

Under certain circumstances the fuel vapor will enter the two-phase region and begin to condense. The decision that such a situation has occurred is made by comparing the pressure $P_{\text{fv,i}}^{n+1}$ with the saturation pressure in cell i. If:

$$P_{\text{fv,i}}^{n+1} < P_{\text{fv,sat,i}}\left(T_{\text{fv,i}}^{n+1}\right) \tag{16.4-114}$$

no condensation will occur during the current time step. Otherwise, some condensation will take place, and the calculation proceeds as outlined below.

First, we determine the temperature $T_{\text{fv,sat,i}}$ where the fuel vapor first reaches saturation. This temperature is obtained using the same method described in Section 16.4.3.5 to correct the sodium temperature where it crosses over from the single phase to the two-phase region. The formula used is similar to Eq. (16.4-73). The enthalpy change between the original temperature $T^n_{\text{fv,i}}$ and the saturation temperature is, by Eq. (16.4-106):

$$\Delta h'_{\text{fv,i}} = \left(T_{\text{fv,sat,i}} - T^n_{\text{fv,i}}\right) * \text{AUXLR} \tag{16.4-115}$$

The total enthalpy change originally calculated is:

$$\Delta h_{\text{fv,i}} = \left(T^{n+1}_{\text{fv,i}} - T^n_{\text{fv,i}}\right) * \text{AUXLR} \tag{16.4-116}$$

The enthalpy which still has to be removed via condensation once the vapor has reached the temperature $T_{\text{fv,sat,i}}$ is given by:

$$\Delta h^{\text{cond}}_{\text{fv,i}} = \Delta h_{\text{fv,i}} - \Delta h'_{\text{fv,i}} = \left(T^{n+1}_{\text{fv,i}} - T_{\text{fv,sat}}\right) * \text{AUXLR} \tag{16.4-117}$$

In order to find the temperature change leading to the enthalpy change $\Delta h^{\text{cond}}_{\text{fv,i}}$, we observe that at the final temperature we have to satisfy the condition for dry vapor:

$$P_{\text{fv,sat}}\left(T_{\text{fv,i}}\right) = R_{\text{fv}} \cdot \rho'_{\text{fv}} \cdot T_{\text{fv,i}} \cdot \frac{1}{\theta_{\text{Mi,i}}} \tag{16.4-118}$$

Also, with the assumption that the heat of vaporization for fuel $h_{\text{fv,lg}}$ is approximately constant for the range of temperatures of interest, the total enthalpy change due to condensation and temperature change can be written as:

$$\Delta h_{\text{fv}} = -\Delta\rho'_{\text{fv,i}} h_{\text{fv,lg}} + 2.5\rho'_{\text{fv,i}} R_{\text{fv}} \cdot \Delta T_{\text{fv}} \tag{16.4-119}$$

where $\Delta\rho'_{\text{fv,i}}$ is the *decrease* in generalized density due to condensation. Because Eq. (16.4-118) and Eq. (16.4-119) cannot be solved directly for the temperature, we use a trial and error approach to find the solution. First, assuming the final temperature if $T^1_{\text{fv}} = T^{n+1}_{\text{fv,i}}$, i.e., the temperature originally calculated, and the amount to condense $\Delta\rho'^1_{\text{fv}}$, Eq. (16.4-118) becomes:

$$P_{\text{fv,sat}}\left(T^1_{\text{fv}}\right) = R_{\text{fv}} \cdot \left(\rho'_{\text{fv,i}} - \Delta\rho'_{\text{fv}}\right) \cdot T^1_{\text{fv}} \cdot \frac{1}{\theta_{\text{Mi}} - \frac{\Delta\rho'_{\text{fv}}}{\rho_{\text{fu,liq}}}} \tag{16.4-120}$$

and, after solving for $\Delta\rho'_{\text{fv}}$:

$$\Delta\rho'^1_{\text{fv}} = \frac{\left(\rho'_{\text{fv,i}} \cdot R_{\text{fv}} \cdot T^1_{\text{fv}} - \theta_{\text{Mi,i}} \cdot P_{\text{fv,sat}}\left(T^1_{\text{fv}}\right)\right)}{R_{\text{fv}} \cdot T^1_{\text{fv}} - \frac{P_{\text{fv,sat}}\left(T^1_{\text{fv}}\right)}{\rho_{\text{fu,liq}}}} \tag{16.4-121}$$

The enthalpy change in Eq. (16.4-119) becomes:

$$\Delta h^1_{\text{fv}} = -\Delta\rho^1_{\text{fv,i}} \cdot h_{\text{hv,lg}} + 2.5 \cdot \rho'_{\text{fv,i}} \cdot R_{\text{fv}} \cdot \left(T^{n+1}_{\text{fv,i}} - T_{\text{fv,sat}}\right) \tag{16.4-122}$$

Then, we assume that the final temperature of the fuel vapor is $T^2_{\text{fv}} = T_{\text{fv,sat}}$, and obtain the new amount of condensate $\Delta\rho'^2_{\text{fv}}$ and the energy change:

$$\Delta h^2_{\text{fv}} = -\Delta\rho'^2_{\text{fv}} \cdot h_{\text{fv,lg}} \tag{16.4-123}$$

These two situations generally will bracket the actual final temperature of the fuel vapor, because the first assumption is practically equivalent to very little or no condensation, while the second assumes maximum condensation

with no temperature change. The actual temperature is obtained by interpolating between T^1_{fv} and T^2_{fv}, with the condition that the final enthalpy change has to be $\Delta h^{\text{cond}}_{\text{fv,i}}$, calculated before from Eq. (16.4-117). Thus:

$$T^{n+1}_{\text{fv,i}} = T^1_{\text{fv,i}} + T^2_{\text{fv,i}} \cdot \frac{\Delta h^{\text{cond}}_{\text{fv,i}} - \Delta h^1_{\text{fv}}}{\Delta h^2_{\text{fv}} - \Delta h^1_{\text{fv}}} \tag{16.4-124}$$

Using this temperature, the actual condensation is calculated from Eq. (16.4-121) and the generalized density of the fuel vapor is updated:

$$\rho'_{\text{fv,i}} = \rho'_{\text{fv,i}} - \Delta\rho'^{\text{cond}}_{\text{fu,i}} \tag{16.4-125}$$

The temperature and generalized density of the liquid fuel is also modified to account for the addition of $\Delta\rho'^{\text{cond}}_{\text{fv,i}}$ at the temperature $T^{n+1}_{\text{fv,i}}$:

$$T_{\text{fu,i}} = \frac{\left[T_{\text{fu,i}}\rho'_{\text{fu,i}} + T^{n+1}_{\text{fv,i}} \cdot \Delta\rho_{\text{fv,i}}\right]}{\left(\rho'_{\text{fu,i}} + \Delta\rho'^{\text{cond}}_{\text{fv,i}}\right)} \tag{16.4-126}$$

$$\rho'_{\text{fu,i}} = \rho'_{\text{fu,i}} + \Delta\rho'^{\text{cond}}_{\text{fv,i}} \tag{16.4-127}$$

16.4.3.6.3 Vaporization of Liquid Fuel

The vaporization model used is a quasi-equilibrium model [16-13] which allows the fuel and fuel vapor to have different temperatures at the same location. Bulk boiling of fuel is assumed to occur whenever the local total pressure is lower than the vapor pressure of the liquid fuel. This process is fast enough to allow a quasi-equilibrium to be established each time step, and is illustrated in Figure 16.4.14 by the original sharp increase in pressure. Thus, the total pressure, including the partial pressure of the fuel vapor becomes equal to the fuel vapor pressure corresponding to the liquid-fuel temperature. When the total pressure exceeds the liquid-fuel vapor pressure, fuel vaporization can occur only by surface vaporization, which is generally a significantly slower process than bulk boiling. The efficiency of surface vaporization is a function of the local flow regime and time-step length. In the limit, if the surface vaporization is assumed to be very efficient, the partial pressure of the fuel vapor can become equal, in each time step, to the liquid-fuel pressure. In this case, the quasi-equilibrium model becomes equivalent to a thermal-equilibrium model.

This calculation is performed for all cells I, with IFFUBT $< I <$ IFFUTP. First, we calculate $P_{\text{fu,sat,i}} = P_{\text{sat}}\left(T_{\text{fu,i}}\right)$. Further vaporization of the liquid fuel in cell I will occur only if:

$$P_{\text{fv,i}} + \left(P_{\text{Na,i}} + P_{\text{fi,i}} + P_{\text{sv,i}}\right) \cdot C_{\text{Pr,fu}} < P_{\text{fu,sat,i}} \tag{16.4-128}$$

where $C_{\text{Pr,fu}}$ is a coefficient dependent on the local configuration and the length of the time step as follows:

If the fuel is the continuous component, i.e., in the annular or bubbly fuel flow regimes

$$C_{Pr,\text{fu}} = 1 \tag{16.4-129}$$

If the fuel is in the form of droplets, i.e. in the annular and bubbly steel flow regimes,

$$C_{\text{Pr,fu}} = \begin{cases} 1 - \Delta t \cdot 10^4 & \text{if } 1 - \Delta t \cdot 10^4 > 0 \\ 0 & \text{if } 1 - \Delta t \cdot 10^4 < 0 \end{cases} \tag{16.4-130}$$

The case $C_{\text{Pr,fu}} = 0$ corresponds to full thermal equilibrium between liquid and fuel vapor.

The vaporization of liquid fuel will take place until the following condition is satisfied:

$$P_{\text{fv,i}} + \left(P_{\text{Na,i}} + P_{\text{fi,i}} + P_{\text{sv,i}}\right) \cdot C_{\text{Pr,fu}} = P_{\text{fu,sat,i}} \tag{16.4-131}$$

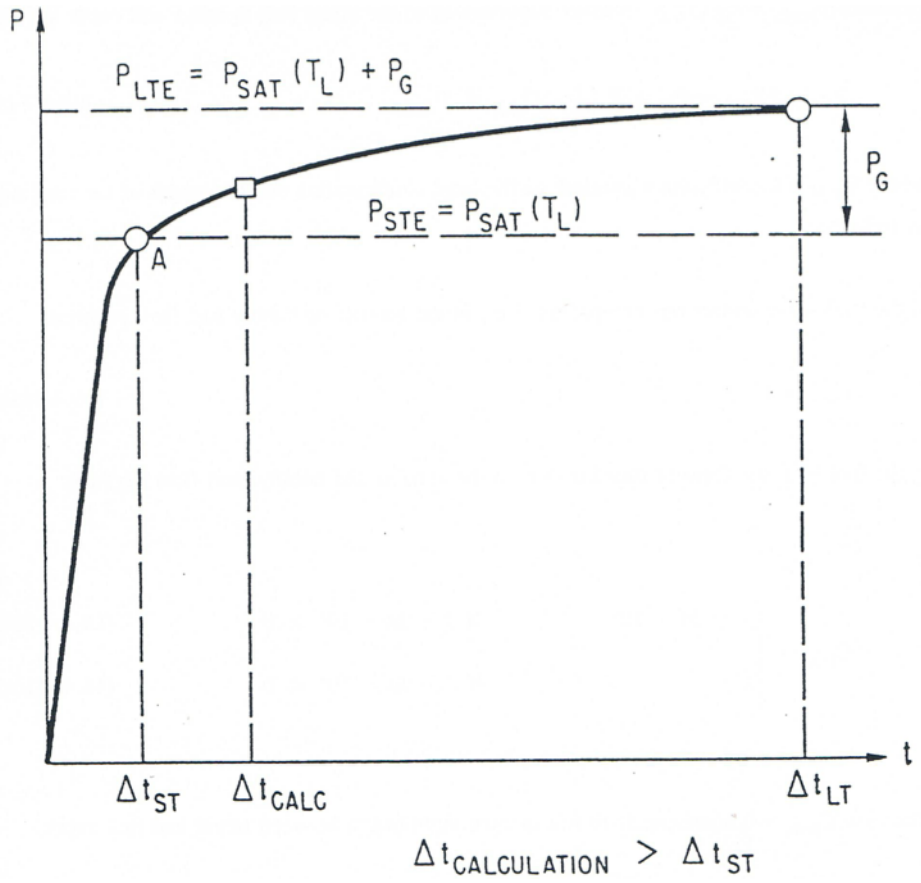


Figure 16.4.14: Time Variation of Fuel Vapor Partial Pressure

Note that all pressures change during vaporization, including $P_{\text{fu,sat,i}}$. Thus, Eq. (16.4-131) cannot be solved directly, and we need to use a trial-and-error approach. First, it is assumed that the final fuel temperature is:

$$T^1_{\text{fu,i}} = T_{\text{fu,i}} \tag{16.4-132}$$

and

$$P^1_{\text{fu,sat,i}} = P_{\text{fu,sat}}\left(T^1_{\text{fu,j}}\right) \tag{16.4-133}$$

We can now calculate the amount of new fuel vapor generated from Eq. (16.4-131):

$$\Delta\rho'_{\text{fv,i}} = \frac{\theta_{\text{Mi,i}}\left[P^1_{\text{fu,sat,i}} - P_{\text{fv,i}} - \left(P_{\text{Na,i}} + P_{\text{fi,i}} + P_{\text{fv,i}}\right)\cdot C_{\text{p,vap,fu}}\right]}{\left[R_{\text{fv}}\cdot T^1_{\text{fv}} - P^1_{\text{fu,sat,i}}\cdot\frac{1}{\rho_{\text{fu,liq}}}\right]} \tag{16.4-134}$$

The new fuel enthalpy is:

$$h'_{\text{fu,i}} = -\frac{\left(h_{\text{fu,i}}\cdot\rho'_{\text{fu,i}} - h_{\text{fu,lq}}\cdot\Delta\rho'_{\text{fv,i}}\right)}{\rho'_{\text{fu,i}} - \Delta\rho'_{\text{fv,i}}} \tag{16.4-135}$$

and the new fuel temperature and vapor pressure are:

$$T'_{\text{fu,i}} = T'_{\text{fu}}\left(h'_{\text{fu,i}}\right) \tag{16.4-136}$$

$$P'_{\text{fu,sat,i}} = P'_{\text{fu,sat}}\left(T_{\text{fu,i}}\right) \tag{16.4-137}$$

A new guess is now made about the final fuel temperature $T^2_{\text{fu,i}} = T'_{\text{fu,i}}$, and the above procedure is repeated, obtaining $P^2_{\text{fu,sat,i}}$, $\Delta\rho''_{\text{fu,i}}$, $T''_{\text{fu,i}}$ and $P''_{\text{fu,sat,i}}$. We can now obtain the real $P_{\text{fu,sat,i}}$, by imposing the condition that the final saturation pressure should be equal to the assumed saturation pressure:

$$P^{n+1}_{\text{fu,sat,i}} = P^i_{\text{fu,sat,i}} - \left(P^2_{\text{fu,sat,i}} - P^1_{\text{fu,sat,i}}\right)\cdot\frac{\Delta P^1}{\Delta P^2 - \Delta P^1} \tag{16.4-138}$$

where

$$\Delta P^1 = P'_{\text{fu,sat,i}} - P^1_{\text{fu,sat,i}} \tag{16.4-139}$$

$$\Delta P^2 = P''_{\text{fu,sat,i}} - P^2_{\text{fu,sat,i}} \tag{16.4-140}$$

Using $P^{n+1}_{\text{fu,sat,i}}$ in Eq. (16.4-134), we can calculate the new $\Delta\rho'_{\text{fv,i}}$. Then, the generalized densities and enthalpies of the fuel vapor and liquid fuel are updated:

$$T^{n+1}_{\text{fv,i}} = \frac{\left(\Delta\rho'_{\text{fv,i}}\cdot T_{\text{fu,i}} + \rho'_{\text{fv,i}}\cdot T_{\text{fv,i}}\right)}{\left(\rho'_{\text{fv,i}} + \Delta\rho'_{\text{fv,i}}\right)} \tag{16.4-141}$$

$$h^{n+1}_{\text{fu,i}} = \frac{\left(h_{\text{fu,i}}\cdot\rho'_{\text{fu,i}} - h_{\text{lq}}\cdot\Delta\rho'_{\text{fv,i}}\right)}{\left(\rho'_{\text{fu,i}} - \Delta\rho'_{\text{fv,i}}\right)} \tag{16.4-142}$$

$$\rho'^{n+1}_{\text{fv,i}} = \rho'_{\text{fv,i}} + \Delta\rho'_{\text{fv,i}} \tag{16.4-143}$$

$$\rho'_{\text{fu,i}} = \rho'_{\text{fu,i}} - \Delta\rho'_{\text{fv,i}} \tag{16.4-144}$$

Finally, the new pressure due to fuel vapor is calculated:

$$P^{n+1}_{\text{fv,i}} = R_{\text{fv}}\cdot\rho'^{n+1}_{\text{fv,i}}\cdot T^{n+1}_{\text{fv,i}}\cdot\frac{1}{\theta^{n+1}_{\text{Mi,i}}} \tag{16.4-145}$$

and is added to the total channel pressure:

$$P^{n+1}_{\text{ch,i}} = P^{n+1}_{\text{ch,i}} + P^{n+1}_{\text{fv,i}} \tag{16.4-146}$$

16.4.3.7 Steel Vapor Energy Conservation Equation

The steel vapor energy conservation equation is solved in the routine LESEVA. This routine is practically identical to the LEFUVA routine, which solves the fuel vapor energy equation and was described in detail in Section 16.4.3.6. At the end of the LESEVA routine, the new pressure due to steel vapor is calculated

$$P_{\text{sv,i}}^{n+1} = R_{\text{sv}} \cdot \rho'^{n+1}_{\text{sv,i}} \cdot T_{\text{sv,i}}^{n+1} \cdot \frac{1}{\theta_{\text{Mi,i}}^{n+1}} \tag{16.4-147}$$

and is added to the total channel pressure:

$$P_{\text{ch,i}}^{n+1} = P_{\text{ch,i}}^{n+1} + P_{\text{sv,i}}^{n+1} \tag{16.4-148}$$

16.4.3.8 Momentum Conservation Equation

The momentum conservation equations are solved in the routine LEMOCO. As indicated previously, the channel hydrodynamic model in LEVITATE calculates three velocity fields, each of them describing the motion of a group of material components, as follows:

$u'_{Mi,i}, u''_{Mi,i}$	the velocity of the gas mixture; described the motion of the two-phase (or superheated) sodium, fission gas, fuel vapor and steel vapor. The code symbol is UMCH.
$u'_{fu,i}, u''_{fu,i}$	the velocity of the liquid fuel and/or liquid steel components. The code symbol is UFCH.
$u_{lu,i}$	the velocity of the solid fuel and/or steel chunks. The code symbol is ULCH.

One is reminded that LEVITATE uses dual velocities to model the motion of the gas mixture and liquid components. Thus, $u'_{\text{Mi,i}}$ represents the velocity of the gas mixture before the $i-1/2$ boundary and $u''_{\text{Mi,i}}$ represents the velocity of the mixture after the same boundary. (Before and after velocities are ordered here by the positive sense of the axial coordinate). As already shown in Section 16.4.3.1, the dual velocities at the boundary $i-1/2$ are related by:

$$C_{\text{Mo,fu,i}} = \frac{u'_{\text{fu,i}}}{u''_{\text{fu,i}}} = \frac{\rho'_{\text{fu,i}}}{\rho'_{\text{fu,i-1}}} \tag{16.4-149}$$

In the code, only the velocity $u''_{\text{fu,i}}$ is stored in the array UFCH (I). The velocity $u'_{\text{fu,i}}$ is always obtained from Eq. (16.4-149), using the coefficient $C_{\text{Mo,fu,i}}$, which is stored in the array CCFU (I). A similar approach is used for the gas mixture dual velocities, which are related at the boundary $i-1/2$ by the coefficients $C_{\text{Mo,Mi,i}}$, stored in the array CCMI (I).

The three momentum equations are solved simultaneously to avoid numerical instabilities due to the generally low inertia of the gas mixture. In cells where only two velocity fields are necessary (e.g., no fuel/steel chunks are present) a system of only two momentum equations is solved. Finally, in cells where the gas mixture only is present, the corresponding momentum equation is solved, while the other two velocity fields remain zero. We first present the derivation of the momentum equations.

16.4.3.8.1 The Momentum Conservation Equation for the Gas Mixture

We begin with the equation written in conservative form for the control volume illustrated in Figure 16.4.15:

$$\frac{\partial}{\partial \mathrm{T}}\left[\rho_{\mathrm{Mi,i\text{-}1}}\cdot\frac{z_{\mathrm{i\text{-}1}}}{2}\cdot A_{\mathrm{Mi,i\text{-}1}}\cdot u'_{\mathrm{Mi,i}}+\rho_{\mathrm{Mi,i}}\cdot\frac{z_i}{2}\cdot A_{\mathrm{Mi,i}}\cdot u''_{\mathrm{Mi,i}}\right]+\left[\left(\rho A u^2\right)_{\mathrm{Mi,i}}-\left(\rho A u^2\right)_{\mathrm{Mi,i\text{-}1}}\right]=$$
$$-A_{\mathrm{Mi,i\text{-}1/2}}\cdot\left(P_{\mathrm{i}}-P_{\mathrm{i\text{-}1}}\right)+\sum_l\left(\Gamma^l_{\mathrm{Mo,i\text{-}1}}\cdot\frac{z_{\mathrm{i\text{-}1}}}{2}+\Gamma^l_{\mathrm{Mo,i}}\cdot\frac{z_{\mathrm{i}}}{2}\right)$$
$$-\rho_{\mathrm{Mi,i\text{-}1}}\cdot\frac{z_{\mathrm{i\text{-}1}}}{2}A_{\mathrm{Mi,i\text{-}1}}\cdot g-\rho'_{\mathrm{Mi,i}}\cdot\frac{z_{\mathrm{i}}}{2}\cdot A_{\mathrm{Mi,i}}\cdot g \tag{16.4-150}$$

where $\sum\Gamma^l_{\mathrm{Mo,i\text{-}1}}$ represents the momentum sources and sinks for the mixture and will be presented in detail later in this chapter. We now divide Eq. (16.4-150) by AXMX and using the definition of the generalized density, the notation:

$$\frac{\Delta z_{\mathrm{i}}}{2}=\Delta z_2\quad;\quad\frac{\Delta z_{\mathrm{i\text{-}1}}}{2}=\Delta z_1 \tag{16.4-151}$$

and the correlation of $u'_{\mathrm{Mi,i}}$ and $u''_{\mathrm{Mi,i}}$, obtain

$$\begin{aligned}&\frac{\partial}{\partial \mathrm{t}}\left[\rho'_{\mathrm{Mi,i\text{-}1}}\cdot\Delta z_{\mathrm{i}}\cdot u_{\mathrm{Mi,i}}\cdot C_{\mathrm{Mo,Mi,i}}+\rho'_{\mathrm{Mi,i}}\cdot\Delta z_2\cdot u_{\mathrm{Mi,i}}\right]\\&=-\left[\left(\rho' u^2\right)_{\mathrm{Mi,i}}-\left(\rho' u^2\right)_{\mathrm{Mi,i\text{-}1}}\right]-\theta_{\mathrm{Mi,i\text{-}1/2}}\cdot\left(P_{\mathrm{i}}-P_{\mathrm{i\text{-}1}}\right)\\&+\sum\left(\Gamma'^l_{\mathrm{Mo,i\text{-}1}}\cdot\Delta z_{\mathrm{i}}+\Gamma'^l_{\mathrm{Mo,i}}\cdot\Delta z_2\right)-\rho'_{\mathrm{Mi,i\text{-}1}}\cdot\Delta z_1\cdot g-\rho'_{\mathrm{Mi,i}}\cdot\Delta z_2 g\end{aligned} \tag{16.4-152}$$

where we used

$$u_{\mathrm{mi,i}}=u''_{\mathrm{Mi,i}} \tag{16.4-153}$$

in order to simplify the notation and:

$$\Gamma'^l_{\mathrm{Mo,i}}=\frac{\Gamma^l_{\mathrm{Mo,i}}}{\mathrm{AXMX}} \tag{16.4-154}$$

The convective fluxes in Eq. (16.4-152) are defined as follows:

$$\left(\rho' u^2\right)_{\mathrm{Mi,i}}=\rho'_{\mathrm{Mi,i}}\cdot\frac{\left(u''_{\mathrm{Mi,i}}+u'_{\mathrm{Mi,i+1}}\right)^2}{4} \tag{16.4-155}$$

An optional formulation of the convective fluxes can be obtained by setting the input parameter IMOMEN $= 1$, in which case:

$$\left(\rho' u^2\right)_{\mathrm{Mi,i}}=\rho'_{\mathrm{Mi,i}}\cdot u^2_{\mathrm{Mo,Mi,i}} \tag{16.4-156}$$

where

$$u_{\mathrm{Mo,Mi,i}}=\begin{cases}u''_{\mathrm{Mi,i}} & \text{if } u''_{\mathrm{Mi,i}}\geq 0 \text{ and } u'_{\mathrm{Mi,i+1}}\geq 0\\ \sqrt{u''^2_{\mathrm{Mi,i}}+u'^2_{\mathrm{Mi,i+1}}} & \text{if } u''_{\mathrm{Mi,i}}\geq 0 \text{ and } u'_{\mathrm{Mi,i+1}}\leq 0\\ 0 & \text{if } u''_{\mathrm{Mi,i}}< 0 \text{ and } u'_{\mathrm{Mi,i+1}}\geq 0\\ u'_{\mathrm{Mi,i+1}} & \text{if } u''_{\mathrm{Mi,i}}< 0 \text{ and } u'_{\mathrm{Mi,i+1}}< 0\end{cases} \tag{16.4-157}$$

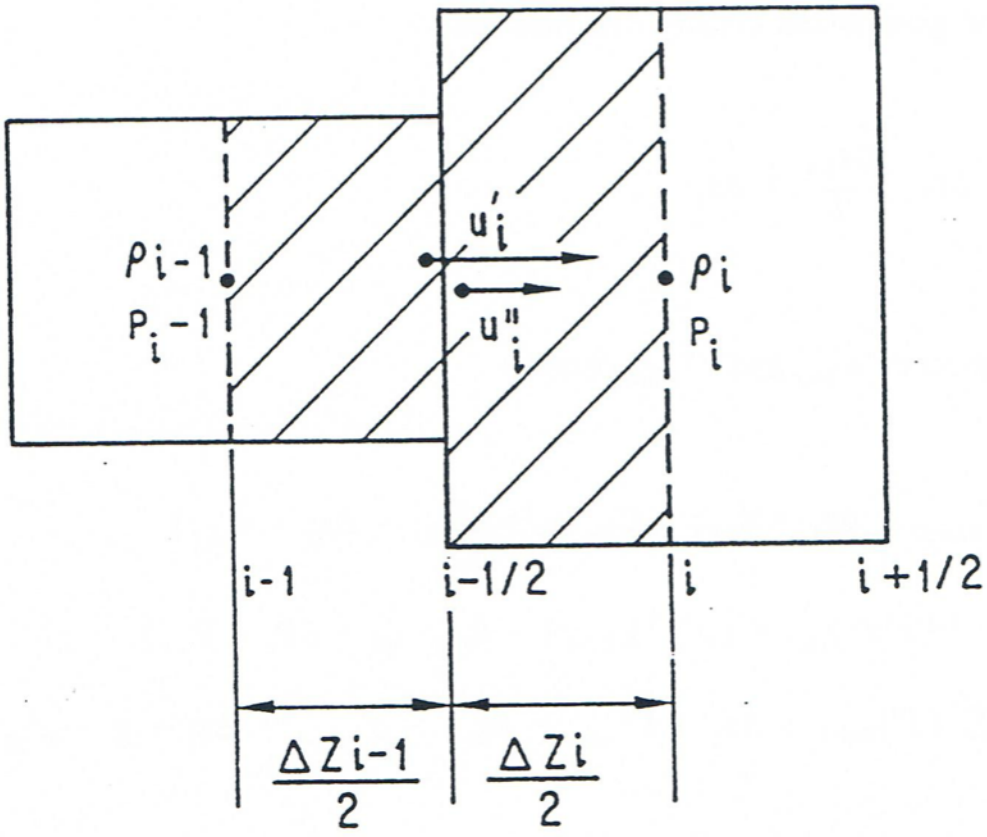


Figure 16.4.15: Control Volume Used for the Solution of the Momentum Equation

This option has been added only recently and has not been tested extensively. It is expected that in future release versions this formulation will become the basic option. The quantity $\theta_{\text{Mi,i-1/2}}$ is defined differently for expansions and contractions [16-5]. For an expansion:

$$\theta_{\text{Mi,i-1/2}} = \begin{cases} \theta_{\text{Mi,i}} & \text{if } u'_{\text{mi,i}} \geq 0 \\ \theta_{\text{Mi,i-1}} & \text{if } u''_{\text{Mi,i}} < 0 \end{cases} \tag{16.4-158}$$

and for contraction:

$$\theta_{\text{Mi,i-1/2}} = C_{\Delta\text{p}} \cdot \frac{\theta_{\text{Mi,i-1}} \cdot \theta_{\text{Mi,i}}}{\theta_{\text{Mi,i-1}} + \theta_{\text{Mi,i}}} \tag{16.4-159}$$

with $C_{\Delta\text{p}}$ currently having the value of 1.67. Eq. (16.4-152) is integrated over the time interval and then divided by Δt. Using the identities:

$$\Delta\left(\rho' u\right) = \rho'^{n+1} \cdot \Delta u + u^n \cdot \Delta\rho' \tag{16.4-160}$$

$$\Delta\left(\rho' u C_{\text{Mo}}\right) = \rho'^{n+1} C_{\text{Mo}}^{n+1} \Delta u + \rho'^{n+1} \cdot u^n \cdot \Delta C_{\text{Mo}} + u^n C^n \cdot \Delta\rho' \tag{16.4-161}$$

we obtain:

$$\begin{aligned} &\Delta u_{\text{Mi,i}} \cdot \left(\rho'^{n+1}_{\text{Mi,i-1}} \cdot C^{n+1}_{\text{Mo,Mi,i}} \cdot \Delta z_{\text{i}} + \rho'^{n+1}_{\text{Mi,i}} \cdot \Delta z_2\right) \cdot \frac{1}{\Delta t} \\ &= -\left[\left(\rho' u^2\right)_{\text{Mi,i}} - \left(\rho' u^2\right)_{\text{Mi,i-1}}\right] - \theta_{\text{Mi,i-1/2}} \cdot \left(P_{\text{i}} - P_{\text{i-1}}\right) \\ &-u_{\text{Mi,i}} \cdot C_{\text{Mo,Mi,i}} \cdot \Delta z_{\text{i}} \cdot \frac{\Delta\rho'_{\text{Mi,i}}}{\Delta t} - u_{\text{Mi,i}} \cdot \Delta z_2 \cdot \frac{\Delta\rho'_{\text{Mi,i}}}{\Delta t} \\ -\rho'^{n+1}_{\text{Mi,i-1}} \cdot u_{\text{Mi,i}} \cdot \Delta z_{\text{i}} \cdot &\frac{\Delta C_{\text{Mo,Mi,i}}}{\Delta t} + \sum_l \left(\Gamma'^{l}_{\text{Mo,Mi,i-1}} \cdot \Delta z_1 + \Gamma'^{l}_{\text{Mo,Mi,i}} \cdot \Delta z_2\right) \\ &-\rho'_{\text{Mi,i-1}} \cdot \Delta z_{\text{i}} \cdot g - \rho'_{\text{Mi,i}} \cdot \Delta z_2 \cdot g \end{aligned} \tag{16.4-162}$$

We will now present the term $\sum_l \Gamma'^{l}_{\text{Mo,i}}$ in more detail:

$$\begin{aligned} \sum_l \Gamma'^{l}_{\text{Mo,Mi,i}} = \Gamma'^{\text{cond}}_{\text{Mo,Mi,i}} + \Gamma'^{\text{vap}}_{\text{Mo,Mi,i}} + \Gamma'^{\text{friction wall}}_{\text{Mo,Mi,i}} + \Gamma'^{\text{drag fuel}}_{\text{Mo,Mi,i}} \\ +\Gamma'^{\text{drag chunks}}_{\text{Mo,Mi},i} + \Gamma'^{\text{inertial}}_{\text{Mo,Mi},i} \end{aligned} \tag{16.4-163}$$

The momentum sink $\Gamma'^{\text{cond}}_{\text{Mo,Mi,i}}$ is due to possible condensation of steel and/or fuel vapor:

$$\Gamma'^{\text{cond}}_{\text{Mo,Mi,i}} = -\frac{\Delta\rho'^{\text{cond}}_{\text{sv,i}}}{\Delta t} u_{\text{Mi,i}} - \frac{\Delta\rho'^{\text{cond}}_{\text{fv,i}}}{\Delta t} \cdot u_{\text{Mi,i}} \tag{16.4-164}$$

Similarly, the vaporization source is defined as:

$$\Gamma'^{\text{vap}}_{\text{Mo,Mi,i}} = \frac{\Delta\rho'^{\text{vap}}_{\text{sv,i}}}{\Delta t} \cdot u_{\text{fu,i}} + \frac{\Delta\rho'^{\text{vap}}_{\text{fv,i}}}{\Delta t} \cdot u_{\text{fu,i}} \tag{16.4-165}$$

The quantities $\Delta\rho'^{\text{cond}}_{\text{fv}}$ and $\Delta\rho'^{\text{vap}}_{\text{fv}}$ are calculated in the routine LEFUVA. These calculations are presented in Section 16.4.3.6. The quantities $\Delta\rho'^{\text{cond}}_{\text{sv}}$ and $\Delta\rho'^{\text{vap}}_{\text{sv}}$ are obtained in the LESEVA routine in the same manner.

The momentum sink due to the wall friction has the form:

$$\begin{aligned} \Gamma'^{\text{friction wall}}_{\text{Mo,Mi,i}} &= -f_{\text{Mi,i}} \cdot \frac{\rho'_{\text{Mi,i}}}{2D_{\text{H,Mi,i}}} \cdot u^{n+1}_{\text{Mi,i}} \cdot \left|u^n_{\text{Mi,i}}\right| \\ &= -f_{\text{Mi,i}} \cdot \frac{\rho'_{\text{Mi,i}}}{2D_{\text{H,Mi,i}}} \cdot \left(u^n_{\text{Mi,i}} + \Delta u_{\text{Mi,i}}\right) \cdot \left|u^n_{\text{Mi,i}}\right| \end{aligned} \tag{16.4-166}$$

The wall friction factor $f_{\text{Mi,i}}$ will be described in Section 16.4.3.10. The hydraulic diameter of the gas mixture $D_{\text{H,Mi,i}}$ will be described in Section 16.4.3.9. The fuel/steel-gas mixture drag source has the form

$$\begin{aligned}\Gamma'^{\text{drag fuel}}_{\text{Mo,Mi,i}} &= C'_{\text{D,Mi,fu,i}}\left(u^{n+1}_{\text{fu,i}} - u^{n+1}_{\text{Mi,i}}\right)\cdot|u_{\text{fi,i}} - u_{\text{Mi,i}}| \\ &= C'_{\text{D,Mi,fu,i}}\cdot(u_{\text{fu,i}} + \Delta u_{\text{fu,i}} - u_{\text{Mi,i}} - \Delta u_{\text{Mi,i}})\cdot|u_{\text{fu,i}} - u_{\text{Mi,i}}|\end{aligned} \tag{16.4-167}$$

The drag coefficient between fuel and mixture has different forms, depending on the local flow regime, as shown below:

For the annular fuel and steel flow regimes:

$$C'_{\text{D,Mi,fu,i}} = f_{\text{Mi,fu,i}}\cdot\frac{\rho'_{\text{Mi,i}}}{2\cdot D_{\text{H,Mi,i}}} \tag{16.4-168}$$

$$C'_{\text{D,Mi,se,i}} = f_{\text{Mi,se,i}}\cdot\frac{\rho'_{\text{Mi,i}}}{2\cdot D_{\text{H,Mi,i}}} \tag{16.4-169}$$

For the bubbly fuel flow regime:

$$C'_{\text{D,Mi,fu,i}} = \theta_{\text{Mi,i}}\cdot\rho_{\text{fu,i}}\cdot\text{CIA6}\cdot\sqrt{\frac{g}{\sigma_{\text{fu}}}\cdot\left(\frac{\rho'_{\text{fu,i}}+\rho'_{\text{se,i}}}{\theta_{\text{fu,i}}+\theta_{\text{se,i}}} - \frac{\rho'_{\text{Mi,i}}}{\theta_{\text{Mi,i}}}\right)}\cdot\left(\frac{\theta_{\text{fu,i}}+\theta_{\text{se,i}}}{\theta_{\text{ch,op,i}}}\right)^2 \tag{16.4-170}$$

where CIA6 is an input constant, currently equal to 0.107. This form of $C'_{\text{D,Mi,fu,i}}$ is explained in more detail in Ch. 14.0, Section 14.4.6.1, which describes the PLUTO2 model.

For the bubbly steel flow regime:

$$C'_{\text{D,Mi,se,i}} = \theta_{\text{Mi,i}}\cdot\rho_{\text{se,i}}\cdot\text{CIA6}\cdot\sqrt{\frac{g}{\sigma_{\text{se}}}\cdot\left(\frac{\rho'_{\text{fu,i}}+\rho'_{\text{se,i}}}{\theta_{\text{fu,i}}+\theta_{\text{se,i}}} - \frac{\rho'_{\text{Mi,i}}}{\theta_{\text{Mi,i}}}\right)}\cdot\left(\frac{\theta_{\text{fu}}+\theta_{\text{se}}}{\theta_{\text{ch,op,i}}}\right)^2 \tag{16.4-171}$$

The factor $f_{\text{Mi,fu,i}}$ which appears in Eq. (16.4-168) will be described in Section 16.4.3.10.

The chunk-mixture drag source has the form:

$$\begin{aligned}\Gamma'^{\text{drag chunk}}_{\text{Mo,Mi,i}} &= C'_{\text{D,Mi,}lu\text{,i}}\cdot\left(u^{n+1}_{lu\text{,i}} - u^{n+1}_{\text{Mi,i}}\right)\cdot|u_{lu\text{,i}} - u_{\text{Mi,i}}| \\ &= C'_{\text{D,Mi,lu,i}}\cdot(u_{lu\text{,i}} + \Delta u_{lu\text{,i}} - u_{\text{Mi,i}} - \Delta u_{\text{Mi,i}})\cdot|u_{lu\text{,i}} - u_{\text{Mi,i}}|\end{aligned} \tag{16.4-172}$$

The drag coefficient $C'_{\text{D,Mi,lu,i}}$ is defined as follows:

For the annular steel or fuel flow regime:

$$C'_{\text{D,Mi,}lu\text{,i}} = \theta_{lu\text{,i}}\cdot\rho_{\text{Mi,i}}\cdot\frac{1}{R_{lu\text{,i}}}\cdot\left(\frac{\theta_{\text{Mi,i}}}{\theta_{\text{ch,i}}}\right)^{\text{CIA5}}\cdot C_{\text{DRAG}}\cdot C_{\text{AREA,Mi,}lu\text{,i}} \tag{16.4-173}$$

where CIA4 and CIA5 are input constants, with the values 0.375 and -2.7, respectively. The coefficient C_{DRAG} is defined as follows:

If $\text{Re}_{\text{lu,i}} > 500$,

$$C_{\text{DRAG}} = 0.44 \tag{16.4-174}$$

If $\text{Re}_{\text{lu,i}} \leq 500$, then

$$C_{\text{DRAG}} = 18.5\cdot(\text{Re}_{lu\text{,i}})^{-6} \tag{16.4-175}$$

The chunk Reynolds number $\text{Re}_{\text{lu,i}}$ used in Eq. (16.4-174) is defined as follows:

$$\text{Re}_{lu\text{,i}} = 2\cdot R_{lu\text{,i}}\cdot|u_{\text{Mi,i}} - u_{lu\text{,i}}|\cdot\rho_{\text{Mi,i}}\cdot\frac{1}{\mu_{\text{Mi,i}}} \tag{16.4-176}$$

The coefficient $C_{\text{AREA,Mi,}l\text{u,i}}$ is used to take into account the fact that the chunks are in contact not only with the gas mixture, but with other components too, such as molten fuel or steel and cladding. This coefficient will be described in Section 16.4.3.10.

For the bubbly steel and fuel flow regimes:

$$C'_{\text{D,Mi,}l\text{u,i}} = 0 \tag{16.4-177}$$

Finally, the inertial (or apparent mass) momentum source term has the form:

$$\Gamma'^{\text{inertial}}_{\text{Mo,Mi,i}} = C'_{\text{IN,Mi,fu,i}} \cdot \left[\frac{\Delta\left(u_{\text{fu,i}} - u_{\text{Mi,i}}\right)}{\Delta t} + u_{\text{Mi,i}} \frac{\Delta\left(u_{\text{fu}} - u_{\text{Mi}}\right)}{\Delta t} \right] \tag{16.4-178}$$

where

$$\frac{\Delta\left(u_{\text{fu}} - u_{\text{Mi}}\right)}{\Delta z} = \begin{cases} \frac{\left(u'_{\text{fu,i}} - u'_{\text{Mi,i}}\right) - \left(u''_{\text{fu,i-1}} - u''_{\text{Mi,i-1}}\right)}{2\Delta z_1} & \text{if } u_{\text{Mi,i}} > 0 \\ \frac{\left(u'_{\text{fu,i+1}} - u'_{\text{Mi,i+1}}\right) - \left(u''_{\text{fu,i}} - u''_{\text{Mi,i}}\right)}{2\Delta z_2} & \text{if } u_{\text{Mi,i}} < 0 \end{cases} \tag{16.4-179}$$

and the generalized inertial coefficient is defined as follows:

For annular steel or fuel flow, no inertial effects are present:

$$C'_{\text{IN,Mi,fu,i}} = 0\ . \tag{16.4-180}$$

For bubbly steel flow:

$$C'_{\text{IN,Mi,fu,i}} = 0.5 \cdot \rho_{\text{se,i}} \cdot \theta_{\text{Mi,i}}\ . \tag{16.4-181}$$

For bubbly fuel flow:

$$C'_{\text{IN,Mi,fu,i}} = 0.5 \cdot \rho_{\text{fu,i}} \cdot \theta_{\text{Mi,i}}\ . \tag{16.4-182}$$

The source terms $\sum \Gamma'^{l}_{\text{Mo,i-1}}$ in Eq. (16.4-163) are similar to the $\sum \Gamma'^{l}_{\text{Mo,i}}$ terms that have been presented in detail, but they *cannot* be obtained from $\Gamma'^{\updownarrow}_{\text{Mo,i}}$ by simply replacing the subscript i by $i-1$. The velocity $u_{\text{Mi,i}}$ in these terms has to be replaced by $u'_{\text{Mi,i}}$, i.e., $C_{\text{Mo,Mi,i}} \cdot u_{\text{Mi,i}}$. The components of $\sum \Gamma'^{l}_{\text{Mo,i-1}}$ are presented below:

$$\Gamma'^{\text{cond}}_{\text{Mo,Mi,i}} = -\frac{\Delta\rho'^{\text{cond}}_{\text{sv,i-1}}}{\Delta t} \cdot C^{n+1}_{\text{Mo,Mi,i}} \cdot u_{\text{Mi,i}} - \frac{\Delta\rho'^{\text{cond}}_{\text{fv,i-1}}}{\Delta t} \cdot C^{n+1}_{\text{Mo,Mi,i}} \cdot u_{\text{Mi,i}} \tag{16.4-183}$$

$$\Gamma'^{\text{vap}}_{\text{Mo,Mi,i-1}} = \frac{\Delta\rho^{\text{vap}}_{\text{sv,i-1}}}{\Delta t} C^{n+1}_{\text{Mo,fu,i}} u_{\text{fu,i}} + \frac{\Delta\rho^{\text{vap}}_{\text{fv,i-1}}}{\Delta t} C^{n+1}_{\text{Mo,fu,i}} \cdot u_{\text{fu,i}} \tag{16.4-184}$$

$$\begin{aligned} \Gamma'^{\text{friction wall}}_{\text{Mo,Mi,i-1}} = -f_{\text{Mi,i-1}} \cdot \frac{\rho'_{\text{Mi,i-1}}}{2 \cdot D_{\text{H,Mi,i-1}}} \cdot \left(u_{\text{Mi,i}} + \Delta u_{\text{Mi,i}}\right) \\ \cdot |u_{\text{MI,i}}| \cdot \left(C^{n+1}_{\text{Mo,Mi,i}}\right)^2 \end{aligned} \tag{16.4-185}$$

$$\begin{aligned} \Gamma'^{\text{drag fuel}}_{\text{Mo,Mi,i-1}} = C'_{\text{D,Mi,fu,i-1}} \left[\left(u_{\text{fu,i}} + \Delta u_{\text{fu,i}}\right) \cdot C^{n+1}_{\text{Mo,fu,i}} - \left(u_{\text{Mi,i}} + \Delta u_{\text{Mi,i}}\right)\right. \\ \left. \cdot C^{n+1}_{\text{Mo,Mi,i}}\right] \cdot \left|u_{\text{fu,i}} \cdot C_{\text{Mo,fu,i}} - u_{\text{Mi,i}} \cdot C_{\text{Mo,Mi,i}}\right| \end{aligned} \tag{16.4-186}$$

$$\begin{aligned} \Gamma'^{\text{drag chunk}}_{\text{Mo,Mi,i-1}} = C'_{\text{D,Mi,}l\text{u,i-1}} \cdot \left[\left(u_{l\text{u,i}} + \Delta u_{l\text{u,i}}\right) - \left(u_{\text{Mi,i}} + \Delta u_{\text{Mi,i}}\right)\right. \\ \left. \cdot C^{n+1}_{\text{Mo,Mi,i}}\right] \cdot \left|u_{l\text{u,i}} - u_{\text{Mi,i}} \cdot C_{\text{Mo,Mi,i}}\right| \end{aligned} \tag{16.4-187}$$

$$\begin{aligned} \Gamma'^{\text{inertial}}_{\text{Mo,Mi,i-1}} = C'_{\text{IN,Mi,fu,i}} \cdot \frac{\Delta\left[\left(u_{\text{fu,i}} \cdot C_{\text{Mo,fu,i}}\right) - \left(u_{\text{Mi,i}} \cdot C_{\text{Mo,Mi,i}}\right)\right]}{\Delta t} \\ + u_{\text{Mi,i}} \cdot C^{n+1}_{\text{Mo,Mi,i}} \cdot \frac{\Delta\left(u_{\text{fu}} - u_{\text{Mi}}\right)}{\Delta z} \end{aligned} \tag{16.4-188}$$

where the definition of $\frac{\Delta(u_{\text{fu}}-u_{\text{Mi}})}{\Delta z}$ is the same as Eq. (16.4-179). It should be noted that, in the definition of the source terms, we used the assumption that changes in the coefficients $C_{\text{Mo,fu,i}}$ can be neglected during one time step, and thus only the new coefficients at time $n+1$ have been used.

We can now replace the expression $\sum_l \left({\Gamma'}^{l}_{\text{Mo,i-1}} \cdot \Delta z_{\text{i}} + {\Gamma'}^{l}_{\text{Mo,i}} \cdot \Delta z_2 \right)$ in Eq. (16.4-163) using Eq. (16.4-164) through Eq. (16.4-186). We then rearrange Eq. (16.4-163) in the form:

$$\text{DMX} \cdot \Delta u_{\text{Mi,i}} = \text{AMX} + \text{BMX} \cdot \Delta u_{\text{fu,i}} + \text{CMX} \cdot \Delta u_{l\text{u,i}} \tag{16.4-189}$$

All terms containing the time change of the gas mixture velocity $\Delta u_{\text{Mi,i}}$ were moved to the left-hand side of the Eq. (16.4-189), and after factoring $\Delta u_{\text{Mi,i}}$, put in the form $\text{DMX} \cdot \Delta u_{\text{Mi,i}}$. Similarly, all terms containing $\Delta u_{\text{fu,i}}$ and $\Delta u_{l\text{u,i}}$ were grouped together on the right-hand side. All other terms were grouped under the coefficient AMX. It should be observed that the term $\frac{\Delta(u_{\text{fu}}-u_{\text{Mi}})}{\Delta z}$ which appears in Eq. (16.4-178) and Eq. (16.4-188) does not contain time changes and thus will be included in the AMX coefficient.

16.4.3.8.2 The Momentum Conservation Equation for the Molten Fuel/Steel Component

Using an integration procedure similar to that used for the gas-mixture equation, we obtain the following equation:

$$\begin{aligned}
\Delta u_{\text{fu,i}} \left[\left({\rho'}^{n+1}_{\text{fu,i-1}} + {\rho'}^{n+1}_{\text{se,i-1}} \right) \cdot C^{n+1}_{\text{Mo,fu,i}} \cdot \Delta z_1 + \left({\rho'}^{n+1}_{\text{fu,}i} + {\rho'}^{n+1}_{\text{se}} \right) \cdot \Delta z_2 \right] \cdot \frac{1}{\Delta t} \\
= - \left[\left(\rho' u^2 \right)_{\text{fu,i}} - \left(\rho' u^2 \right)_{\text{fu,i}} \right] - \theta_{\text{fu,i-1/2}} \cdot (P_{\text{I}} - P_{\text{i-1}}) \\
- u_{\text{fu,i}} \cdot C_{\text{Mo,fu,i}} \cdot \Delta z_{\text{i}} \cdot \left(\frac{\Delta {\rho'}_{\text{fu,i-1}}}{\Delta t} + \frac{\Delta {\rho'}_{\text{se,i-1}}}{\Delta t} \right) \\
- u_{\text{fu,i}} \cdot \Delta z_2 \cdot \left(\frac{\Delta {\rho'}_{\text{fu,i}}}{\Delta t} + \frac{\Delta {\rho'}_{\text{se,i}}}{\Delta t} \right) - \left({\rho'}^{n+1}_{\text{fu,i-1}} + {\rho'}^{n+1}_{\text{se,i-1}} \right) \cdot u_{\text{fu,i}} \cdot \Delta z_{\text{i}} \cdot \frac{\Delta C_{\text{Mo,fu,i}}}{\Delta t} \\
+ \sum_{\text{m}} \left({\Gamma'}^{m}_{\text{Mo,fu,i-1}} \cdot \Delta z_1 + {\Gamma'}_{\text{Mo,fu,i}} \cdot \Delta z_2 \right) - \left({\rho'}_{\text{fu,i-1}} + {\rho'}_{\text{se,i-1}} \right) \cdot \Delta z_1 \cdot g \\
- \left({\rho'}_{\text{fu,i}} + {\rho'}_{\text{se,i}} \right) \cdot \Delta z_2 \cdot g
\end{aligned} \tag{16.4-190}$$

where

$$\left(\rho' u^2 \right)_{\text{fu,i}} = \left({\rho'}_{\text{fu,i}} + {\rho'}_{\text{se,i}} \right) \cdot \left({u''}_{\text{fu,i}} + {u'}_{\text{fu,i+1}} \right)^2 \cdot .25 \tag{16.4-191}$$

The optional formulation, which can be obtained using IMOMEN $= 1$ is:

$$\left(\rho' u^2 \right)_{\text{fu,i}} = \left({\rho'}_{\text{fu,i}} + {\rho'}_{\text{se,i}} \right) \cdot u^2_{\text{Mo,fu,i}} \tag{16.4-192}$$

where

$$u_{\text{Mo,Mi,i}} = \begin{cases} {u''}_{\text{fu,i}} & \text{if } {u''}_{\text{fu,i}} \geq 0 \text{ and } {u'}_{\text{fu,i+1}} \geq 0 \\ \sqrt{{u''}^2_{\text{fu,i}} + {u'}^2_{\text{fu,i+1}}} & \text{if } {u''}_{\text{fu,i}} \geq 0 \text{ and } {u'}_{\text{fu,i+1}} \leq 0 \\ 0 & \text{if } {u''}_{\text{fu,i}} < 0 \text{ and } {u'}_{\text{fu,i+1}} \geq 0 \\ {u'}_{\text{fu,i+1}} & \text{if } {u'}_{\text{fu,i}} < 0 \text{ and } {u'}_{\text{fu,i+1}} < 0 \end{cases} \tag{16.4-193}$$

The quantity $\theta_{\text{fu,i-1/2}}$ is defined by Eq. (16.4-157) and 16.4-143, where the $\theta_{\text{Mi,i}}$ is replaced by $(\theta_{\text{fu,i}} + \theta_{\text{se,i}})$. The source/sink terms in $\sum_{\text{m}} {\Gamma'}^{m}_{\text{Mo,fu,i}}$ are defined in detail below:

The condensation term:

$${\Gamma'}^{\text{cond}}_{\text{Mo,fu,i}} = \frac{\Delta {\rho'}^{\text{cond}}_{\text{fv,i}}}{\Delta t} \cdot u_{\text{Mi,i}} \tag{16.4-194}$$

The vaporization term:

$$\Gamma'^{\text{vap}}_{\text{Mo,fu,i}} = -\frac{\Delta\rho'_{\text{fv,i}}}{\Delta t} \cdot u_{\text{fu,i}} \tag{16.4-195}$$

The wall friction term is dependent on the flow regime:

For the annular or bubbly fuel flow regimes:

$$\begin{aligned}\Gamma'^{\text{friction wall}}_{\text{Mo,fu,i}} = &-f_{\text{se,i}} \cdot \frac{\rho_{\text{se,i}} \cdot (\theta_{\text{fu,i}} + \theta_{\text{se,i}})}{2 \cdot D_{\text{H,se,i}}} \cdot C_{\text{AREA,se,i}} \cdot u^{n+1}_{\text{fu,i}} \cdot \left|u^{n}_{\text{fu,i}}\right| \\ &-f_{\text{se,i}} \frac{\rho_{\text{se,i}} (\theta_{\text{fu,i}} + \theta_{\text{se,i}})}{2D_{\text{H,se,i}}} \cdot C_{\text{AREA,se,i}} \cdot (u_{\text{fu,i}} + \Delta u_{\text{fu,i}}) \cdot |u_{\text{fu,i}}|\end{aligned} \tag{16.4-196}$$

For the annular or bubbly fuel flow regimes:

$$\Gamma'^{\text{friction wall}}_{\text{Mo,fu,i}} = -f_{\text{fu,i}} \cdot \frac{\rho_{\text{fu,i}} (\theta_{\text{fu,i}} + \theta_{\text{se,i}})}{2D_{\text{H,fu,i}}} \cdot C_{\text{AREA,fu,i}} \cdot (u_{\text{fu,i}} + \Delta u_{\text{fu,i}}) \cdot |u_{\text{fu,i}}| \tag{16.4-197}$$

where

the $D_{\text{H,se,i}}$ and $D_{\text{H,fu,i}}$ are the hydraulic diameters for steel and fuel, respectively, and are described in Section 16.4.3.9.

the contact coefficients $C_{\text{AREA,se,i}}$ and $C_{\text{AREA,fu,i}}$ account for the fact that only a fraction of the steel or fuel perimeter is in contact with the stationary walls; they are described in Section 16.4.3.10; and the friction coefficients $f_{\text{se,i}}$ and $f_{\text{fu,i}}$ are defined below:

$$f_{\text{se,i}} = \begin{cases} \text{CIFRFU} & \text{if } \text{Re}_{\text{se,i}} \geq \text{CIREFU} \\ \frac{80}{Re_{\text{se,i}}} & \text{if } \text{Re}_{\text{se,i}} < \text{CIREFU} \end{cases} \tag{16.4-198}$$

$$f_{\text{fu,i}} = \begin{cases} \text{CIFRFU} & \text{if } \text{Re}_{\text{fu,i}} \geq \text{CIREFU} \\ \frac{80}{\text{Re}_{\text{fu,i}}} & \text{if } \text{Re}_{\text{fu,i}} < \text{CIREFU} \end{cases} \tag{16.4-199}$$

where CIFRFU and CIREFU are input constants. Currently, CIFRFU $= .03$ and CIREFU $= 2100$.

The fuel/steel-gas mixture drag source is similar, but of opposite sign, to the term already presented for the gas mixture.

$$\begin{aligned}\Gamma'^{\text{drag mixture}}_{\text{Mo,fu,i}} = -C'_{\text{D,Mi,fu,i}} \cdot (u_{\text{fu,i}} + \Delta u_{\text{fu,i}} - u_{\text{Mi,i}} - \Delta u_{\text{Mi,i}}) \\ \cdot |u_{\text{fu,i}} - u_{\text{Mi,i}}|\end{aligned} \tag{16.4-200}$$

The drag coefficient $C'_{\text{D,Mi,fu,i}}$ is defined by Eq. (16.4-168) and Eq. (16.4-170).

The fuel/steel - chunk drag source is defined as follows:

$$\begin{aligned}\Gamma'^{\text{drag chunk}}_{\text{Mo,fu,i}} &= C'_{\text{D,fu,}l\text{u,i}} \cdot \left(u^{n+1}_{l\text{u,i}} - u^{n+1}_{\text{fu,i}}\right) \cdot \left|u^{n}_{l\text{u,i}} - u^{n}_{\text{fu,i}}\right| \\ &= C'_{\text{D,fu,}l\text{u,i}} \cdot (u_{l\text{u,i}} + \Delta u_{l\text{u,i}} - u_{\text{fu,i}} - \Delta u_{\text{fu,i}}) \cdot |u_{l\text{u,i}} - u_{\text{fu,i}}|\end{aligned} \tag{16.4-201}$$

The generalized drag coefficient $C'_{\text{D,fu,}l\text{u,i}}$ is flow regime dependent and is defined below:

For the steel annular and bubbly flow regimes:

$$C'_{\text{D,fu,}l\text{u,i}} = \theta_{l\text{u,i}} \cdot \rho_{\text{se,i}} \cdot \text{CIA4} \cdot \frac{1}{R_{l\text{u,i}}} \cdot \left[\frac{\theta_{\text{Mi,i}}}{\theta_{\text{Mi,i}} + \theta_{l\text{u,i}}}\right]^{\text{CIA5}} \cdot C_{\text{DRAG}} \cdot C_{\text{AREA,se,}l\text{u,i}} \tag{16.4-202}$$

For the fuel annular and bubbly flow regimes:

$$C'_{\text{D,fu,}l\text{u,i}} = \theta_{l\text{u,i}} \cdot \rho_{\text{fu,i}} \cdot \text{CIA4} \cdot \frac{1}{R_{l\text{u,i}}} \cdot \left[\frac{\theta_{\text{Mi,i}}}{\theta_{\text{Mi,i}} + \theta_{l\text{u,i}}}\right]^{\text{CIA5}} \cdot C_{\text{DRAG}} \cdot C_{\text{AREA,fu,}l\text{u,i}} \tag{16.4-203}$$

In the above equations, the drag coefficient C_{DRAG} is defined by 16.4-156. The area coefficients, which account for the fact that only a fraction of the chunk lateral area is in contact with the molten fuel or steel, are defined below:

$$C_{\text{AREA,se,}l\text{u,i}} = \begin{cases} (1 - C_{\text{AREA,Mi,}l\text{u,i}}) \cdot C_{\text{AREA,se,}cl\text{,i}} & \text{for annular steel flow} \\ 1 & \text{for bubbly steel flow} \end{cases} \tag{16.4-204}$$

where $C_{\text{AREA,Mi,}l\text{u,i}}$ indicates the fraction of the lateral chunk area in contact with the gas mixture. The quantity $1 - C_{\text{AREA,Mi,}l\text{u,i}}$ thus represents the fraction of the chunk area in contact with the molten fuel/steel and cladding/hexcan wall. The coefficient $C_{\text{AREA,se,cl}}$ represents the fraction of the cladding/hexcan wall area covered by molten steel/fuel. Both these area coefficients will be explained in more detail in Section 16.4.3.9.

The momentum source due to fuel injection from the pin

$$\Gamma'^{\text{injection}}_{\text{Mo,fu,i}} = \frac{\Delta\rho'_{\text{fu,injection,i}}}{\Delta t} \cdot u_{\text{fu,ca,i}} \cdot C_{\text{Mo,injection,i}} \tag{16.4-205}$$

The quantity $\Delta\rho'_{\text{fu,injection,i}}$ represents the change in the fuel generalized density due to injection via the pin rip or via the end of the pin stubs, as explained in Section 16.3. The velocity of the injected material is $u_{\text{fu,ca,i}}$, which is also explained in Section 16.3. The coefficient $C_{\text{Mo,injection,i}}$ accounts for the axial momentum loss due to lateral acceleration and mixing during the injection process, and is defined as follows:

$$C_{\text{Mo,injection,i}} = \begin{cases} 1, & \text{if injection is taking place via the pin stubs} \\ \text{CIFUMO}, & \text{if injection is taking place via the pin rip} \end{cases} \tag{16.4-206}$$

where CIFUMO is an input constant, with values between 0 and 1.

To summarize, the term $\sum_{\text{m}} \Gamma'^{m}_{\text{Mo,fu,i}}$, which appears in Eq. (16.4-190), has the form,

$$\begin{aligned} \sum_{\text{m}} \Gamma'^{m}_{\text{Mo,fu,i}} = \Gamma'^{\text{cond}}_{\text{Mo,fu,i}} + \Gamma'^{\text{vap}}_{\text{Mo,fu,i}} + \Gamma'^{\text{friction wall}}_{\text{Mo,fu,i}} + \Gamma'^{\text{drag mixture}}_{\text{Mo,fu,i}} \\ + \Gamma'^{\text{drag chunk}}_{\text{Mo,fu,i}} + \Gamma'^{\text{injection}}_{\text{Mo,fu,i}} \end{aligned} \tag{16.4-207}$$

The term $\sum_{\text{m}} \Gamma'^{m}_{\text{Mo,fu,i-1}}$ has a similar composition, and its terms are presented below:

$$\Gamma'^{\text{cond}}_{\text{Mo,fu,i-1}} = \frac{\Delta\rho'^{\text{cond}}_{\text{fv,i-1}}}{\Delta t} u_{\text{Mi,i}} \cdot C_{\text{Mo,Mi,i}} \tag{16.4-208}$$

$$\Gamma'^{\text{vap}}_{\text{Mo,fu,i-1}} = -\frac{\Delta\rho'_{\text{fv,i-1}}}{\Delta t} \cdot u_{\text{fu,i}} \cdot C_{\text{Mo,fu,i}} \tag{16.4-209}$$

For the steel flow regimes,

$$\begin{aligned} \Gamma'^{\text{friction}}_{\text{Mo,fu,i-1}} = -f_{\text{se,i-1}} \cdot \frac{\rho_{\text{se,i-1}} \cdot (\theta_{\text{fu,i-1}} + \theta_{\text{se,i-1}})}{2D_{\text{H,se,i-1}}} \cdot C_{\text{AREA,se,i-1}} \\ \cdot C^2_{\text{Mo,fu,i}} \cdot (u_{\text{fu,i}} + \Delta u_{\text{fu,i}}) \, |u_{\text{fu,i}}| \end{aligned} \tag{16.4-210}$$

For the fuel flow regimes:

$$\begin{aligned} \Gamma'^{\text{friction}}_{\text{Mo,fu,i-1}} = -f_{\text{fu,i-1}} \cdot \frac{\rho_{\text{fu,i-1}} \cdot (\theta_{\text{fu,i-1}} + \theta_{\text{se,i-1}})}{2D_{\text{H,fu,i-1}}} \cdot C_{\text{AREA,fu,i-1}} \\ \cdot C^2_{\text{Mo,fu,i}} \cdot (u_{\text{fu,i}} + \Delta u_{\text{fu,i}}) \, |u_{\text{fu,i}}| \end{aligned} \tag{16.4-211}$$

$$\Gamma'^{\text{drag mixture}}_{\text{Mo,fu,i-1}} = -C'_{\text{D,Mi,fu,i-1}} \cdot \left[(u_{\text{fu,i}} + u_{\text{fu,i}}) \cdot C^{n+1}_{\text{Mo,fu,i}} - (u_{\text{Mi,i}} + u_{\text{Mi,i}}) \cdot C^{n+1}_{\text{Mo,Mi,i}} \right] \cdot |u_{\text{fu,i}} \cdot C_{\text{Mo,fu,i}} - u_{\text{Mi,i}} C_{\text{Mo,fu,i}}| \tag{16.4-212}$$

$$\Gamma'^{\text{drag chunk}}_{\text{Mo,fu,i-1}} = C'_{\text{D,fu,}lu\text{,i-1}} \left[(u_{lu,\text{i}} + \Delta u_{lu,\text{i}}) - (u_{\text{fu,i}} + \Delta u_{\text{fu,i}}) \cdot C^{n+1}_{\text{Mo,fu,i}} \right] \cdot |u_{lu,\text{i}} - u_{\text{fu,i}} \cdot C_{\text{Mo,fu,i}}| \tag{16.4-213}$$

$$\Gamma'^{\text{injection}}_{\text{Mo,fu,i-1}} = \frac{\Delta \rho'_{\text{fu,injection,i-1}}}{\Delta t} \cdot u_{\text{fu,ca,i-1}} \cdot C_{\text{Mo,injection,i-1}} \tag{16.4-214}$$

The source terms, as given by Eq. (16.4-194) through Eq. (16.4-214) are substituted in Eq. (16.4-190), and, after rearranging, the fuel momentum equation is written in the form

$$\text{DFX} \cdot \Delta u_{\text{fu}} = \text{AFX} + \text{BFX} \cdot \Delta u_{\text{Mi}} + \text{CFX} \cdot \Delta u_{\text{fu}} \tag{16.4-215}$$

16.4.3.8.3 The Momentum Conservation Equation for the Fuel/Steel Chunks

Using an integration procedure similar to that used for the gas mixture, we obtain the following equation in finite difference form:

$$\begin{aligned} \Delta u_{lu,\text{i}} \left[\left(\rho'^{n+1}_{fl,\text{i-1}} + \rho'^{n+1}_{sl,\text{i-1}} \right) \cdot \Delta z_{\text{i}} + \left(\rho'^{n+1}_{fl,\text{i-1}} + \rho'^{n+1}_{sl,\text{i-1}} \right) \cdot \Delta z_2 \right] \cdot \frac{1}{\Delta t} \\ = - \left[\left(\rho' u^2 \right)_{lu,\text{i}} - \left(\rho' u^2 \right)_{lu,\text{i-1}} \right] - \theta_{lu,\text{i-1/2}} \cdot (P_{\text{i}} - P_{\text{i-1}}) \\ - u_{lu,\text{i}} \cdot \Delta z_1 \cdot \left(\frac{\Delta \rho'_{fl,\text{i-1}}}{\Delta t} + \frac{\Delta \rho'_{sl,\text{i-1}}}{\Delta t} \right) - u_{lu,\text{i}} \cdot \Delta z_2 \cdot \left(\frac{\Delta \rho'_{fl,\text{i}}}{\Delta t} + \frac{\Delta \rho'_{sl,\text{i}}}{\Delta t} \right) \\ + \sum_{\text{n}} \left(\Gamma'^{n}_{\text{Mo,}lu,\text{i}} \cdot \Delta z_{\text{i}} + \Gamma'^{n}_{\text{Mo,}lu,\text{i}} \cdot \Delta z_2 \right) - \left(\rho'_{fl,\text{i-1}} + \rho'_{sl,\text{i-1}} \right) \cdot \Delta z_1 \cdot g \\ - \left(\rho'_{fl,\text{i}} + \rho'_{sl,\text{i}} \right) \cdot \Delta z_2 \cdot g \end{aligned} \tag{16.4-216}$$

where

$$\left(\rho' u^2 \right)_{lu,\text{i}} = \left(\rho'_{fl,\text{i}} + \rho'_{sl,\text{i}} \right) \cdot \left(u_{lu,\text{i}} + u_{lu,\text{i+1}} \right)^2 \cdot 0.25 \tag{16.4-217}$$

The optimal formulation of the convective terms can be obtained by using the input variable IMOMEN $= 1$ and is similar to Eq. (16.4-193), where $u''_{\text{fu,i}}$ and $u'_{\text{fu,i}}$ are replaced by $u_{lu,\text{i}}$ and $u_{lu,\text{i}}$ and $u_{lu,\text{i+1}}$, respectively. The quantity $\theta_{lu,\text{i-1/2}}$ is defined by Eq. (16.4-158) and 16.4-144, where $\theta_{\text{Mi,i}}$ is replaced by $\theta_{lu,\text{i}}$. The source terms are defined below:

$$\sum_{\text{n}} \Gamma'^{n}_{\text{Mo,}lu,\text{i}} = \Gamma'^{\text{drag mixture}}_{\text{Mo,}lu,\text{i}} + \Gamma'^{\text{drag fuel}}_{\text{Mo,}lu,\text{i}} + \Gamma'^{\text{friction wall}}_{\text{Mo,lu,i}} + \Gamma'^{\text{contraction}}_{\text{Mo,}lu,\text{i}} \tag{16.4-218}$$

No momentum sources due to chunk formation or remelting are present in Eq. (16.4-218), because the routines modeling these processes are called after the LEMOCO routine, and thus these effects will be considered later. The individual sources are presented below:

The term due to the gas mixture/chunk drag:

$$\Gamma'^{\text{drag mixture}}_{\text{Mo,}lu,\text{i}} = -C'_{\text{D,Mi,lu,i}} \cdot (u_{lu,\text{i}} + \Delta u_{lu,\text{i}} - u_{\text{Mi,i}} - \Delta u_{\text{Mi,i}}) \cdot |u_{lu,\text{i}} + u_{\text{Mi,i}}| \tag{16.4-219}$$

The generalized drag coefficient $C'_{\text{D,Mi,}lu,\text{i}}$ has been presented in Eq. (16.4-201).

The term due to the molten fuel/steel-chunk drag is as follows:

$$\Gamma'^{\text{drag fuel}}_{\text{Mo,}lu,\text{i}} = -C'_{\text{D,fu,}lu,\text{i}} \cdot (u_{lu,\text{i}} + \Delta u_{lu,\text{i}} - u_{\text{fu,i}} - \Delta u_{\text{fu,i}}) \cdot |u_{lu,\text{i}} + u_{\text{fu,i}}| \tag{16.4-220}$$

The generalized drag coefficient $C'_{\text{D,fu,}lu,\text{i}}$ has been presented in Eq. (16.4-201).

The momentum sink due to wall friction is:

$$\Gamma'^{\text{friction wall}}_{\text{Mo},lu,\text{i}} = -0.5 \cdot 10^{-3} \cdot P_{\text{i}} \cdot A'_{lu,\text{cl,i}} \cdot \text{SIGN}\left(u_{lu,\text{i}}\right)$$

where the generalized area of contact between chunks and clad/hexcan wall, $A'_{lu,\text{cl,i}}$ is defined as:

$$A'_{lu,\text{cl,i}} = \frac{A_{lu,\text{cl,i}}}{\Delta z_{\text{i}} \cdot \text{AXMX}} \tag{16.4-221}$$

and is presented in Section 16.4.3.9. Eq. (16.4-220) was obtained assuming that the frictional force between chunks and wall is due to the normal force generated by the pressure P_{i} and that the friction coefficient between the two solid surfaces is $0.5 \cdot 10^{-3}$.

The momentum sink due to jumbling at the contraction at the boundary i is due to chunks arriving at an abrupt contraction, where they lose momentum upon hitting the wall normal to the flow path:

If $u_{lu,\text{i}} < 0$ and $A_{\text{i}} > A_{\text{i-1}}$,

$$\Gamma'^{\text{jumbling}}_{\text{Mo},lu,\text{i}} = \left(\rho'_{fl,\text{i}} + \rho'_{sl,\text{i}}\right) \cdot u^2_{lu,\text{i}} \cdot \frac{A_{\text{i}} - A_{\text{i-1}}}{A_{\text{i}}} \tag{16.4-222}$$

Otherwise,

$$\Gamma'^{\text{jumbling}}_{\text{Mo},lu,\text{i}} = 0 \tag{16.4-223}$$

The sources $\sum_{\text{n}} \Gamma'^{n}_{\text{Mo},lu,\text{i-1}}$ are defined in a similar way, as shown below:

$$\begin{aligned}\Gamma'^{\text{drag mixture}}_{\text{Mo},lu,\text{i-1}} = -C'_{\text{D,Mi},lu,\text{i-1}} \cdot \big[&(u_{lu,\text{i}} + \Delta u_{lu,\text{i}}) - (u_{\text{Mi,i}} + \Delta u_{\text{Mi,i}}) \\ &\cdot C^{n+1}_{\text{Mo,Mi,i}}\big] \cdot \left|u_{lu,\text{i}} - u_{\text{Mi,i}} \cdot C_{\text{Mo,Mi,i}}\right|\end{aligned} \tag{16.4-224}$$

$$\begin{aligned}\Gamma'^{\text{drag fuel}}_{\text{Mo},lu,\text{i-1}} = -C'_{\text{D,fu},lu,\text{i-1}} \cdot \big[&(u_{lu,\text{i}} + \Delta u_{lu,\text{i}}) - (u_{\text{fu,i}} + \Delta u_{\text{fu,i}}) \\ &\cdot C^{n+1}_{\text{Mo,fu,i}}\big] \cdot \left|u_{lu,\text{i}} - u_{\text{fu,i}} \cdot C_{\text{Mo,fu,i}}\right|\end{aligned} \tag{16.4-225}$$

$$\Gamma'^{\text{friction wall}}_{\text{Mo},lu,\text{i-1}} = -0.5 \cdot 10^{-3} \cdot P_{\text{i-1}} \cdot A'_{lu,\text{cl,i-1}} \cdot \text{SIGN}\left(u_{lu,\text{i}}\right) \tag{16.4-226}$$

If $u_{lu,\text{i}} \geq 0$ and $A_{\text{i-1}} > A_{\text{i}}$,

$$\Gamma'^{\text{contraction}}_{\text{Mo},lu,\text{i-1}} = \left(\rho'_{fl,\text{i-1}} + \rho'_{sl,\text{i-1}}\right) \cdot u^2_{lu,\text{i}} \cdot \frac{A_{\text{i-1}} - A_{\text{i}}}{A_{\text{i-1}}} \tag{16.4-227}$$

Otherwise,

$$\Gamma'^{\text{contraction}}_{\text{Mo},lu,\text{i-1}} = 0 \tag{16.4-228}$$

The source terms given by Eq. (16.4-219) through Eq. (16.4-227) are substituted in Eq. (16.4-216) and after rearranging, we obtain the chunk momentum equation

$$\text{DLX} \cdot \Delta u_{lu,\text{i}} = \text{ALX} + \text{BLX} \cdot \Delta u_{\text{fu,i}} + \text{CLX} \cdot \Delta u_{\text{Mi,i}} \tag{16.4-229}$$

16.4.3.8.4 The Simultaneous Solution of the Momentum Conservation Equations

The calculation of the new velocities in momentum cell i begins by calculating the coefficients AMX and DMX for the gas-mixture Eq. (16.4-189). The terms due to the mixture interaction with the molten fuel/steel and

chunks are not included in AMX and DMX at this time. A check is then performed to verify if the momentum cell I contains only the gas mixture. If this is the case, the only equations solved is

$$\text{DMX} \cdot \Delta u_{\text{Mi,i}} = \text{AMX} \tag{16.4-230}$$

and only the gas-mixture velocity is updated:

$$u_{\text{Mi,i}}^{n+1} = u_{\text{Mi,i}}^{n} + \Delta u_{\text{Mi,i}} \tag{16.4-231}$$

If either half of the momentum cell i contains molten fuel and/or steel, the coefficients are calculated for the momentum equations:

$$\text{DMX} \cdot \Delta u_{\text{Mi,i}} = \text{AMX} + \text{BMX} \cdot \Delta u_{\text{fu,i}} \tag{16.4-232}$$

$$\text{DFX} \cdot \Delta u_{\text{fu,i}} = \text{AFX} + \text{BFX} \cdot \Delta u_{\text{Mi,i}} \tag{16.4-233}$$

If the momentum cell does not contain fuel/steel chunks, these equations are solved simultaneously for $\Delta u_{\text{Mi,i}}$ and $\Delta u_{\text{fu,i}}$ and the new velocities are calculated:

$$u_{\text{Mi,i}}^{n+1} = u_{\text{Mi,i}}^{n} + \Delta u_{\text{Mi,i}} \tag{16.4-234}$$

$$u_{\text{fu,i}}^{n+1} = u_{\text{fu,i}}^{n} + \Delta u_{\text{fu,i}} \tag{16.4-235}$$

It is noted that the coefficients AMX and DMX in Eq. (16.4-232) are obtained by adding to the values calculated for Eq. (16.4-230) the additional terms due to the gas mixture-fuel interaction. If either half of the momentum cell contains fuel/steel chunks, the coefficients are calculated for the momentum equations in the form below:

$$\text{DMX} \cdot \Delta u_{\text{Mi,i}} = \text{AMX} + \text{BMX} \cdot \Delta u_{\text{fu,i}} + \text{CMX} \cdot \Delta u_{l\text{u,i}} \tag{16.4-236}$$

$$\text{DFX} \cdot \Delta u_{\text{fu,i}} = \text{AFX} + \text{BFX} \cdot \Delta u_{\text{Mi,i}} + \text{CFX} \cdot \Delta u_{l\text{u,i}} \tag{16.4-237}$$

$$\text{DLX} \cdot \Delta u_{l\text{u,i}} = \text{ALX} + \text{BLX} \cdot \Delta u_{\text{fu,i}} + \text{CLX} \cdot \Delta u_{\text{Mi,i}} \tag{16.4-238}$$

The coefficients that have been calculated using for Eq. (16.4-234) are updated by adding the terms due to the presence of chunks. Eq. (16.4-236) are solved simultaneously for $\Delta u_{\text{Mi,i}}, \Delta_{\text{fu,i}}, \Delta u_{l\text{u,i}}$, by using a substitution method and the new velocities are calculated:

$$u_{\text{Mi,i}}^{n+1} = u_{\text{Mi,i}}^{n} + \Delta u_{\text{Mi,i}} \tag{16.4-239}$$

$$u_{\text{fu,i}}^{n+1} = u_{\text{fu,i}}^{n} + \Delta u_{\text{fu,i}} \tag{16.4-240}$$

$$u_{l\text{u,i}}^{n+1} = u_{l\text{u,i}}^{n} + \Delta u_{l\text{u,i}} \tag{16.4-241}$$

16.4.3.9 Description of the Local Geometry

In order to fully define the energy and momentum source terms used in the conservation equations, we must supply the areas of contact between various components. These areas are obtained in LEVITATE by defining the local geometry in the routine LEGEOM. The definition of the geometry is based on the local flow regime and the configuration of the stationary elements, i.e., presence of fuel pins, presence of cladding, presence of frozen fuel crusts, etc.

16.4.3.9.1 Local Flow Regime Definition

Because the flow regimes are important in defining the local geometry, the decision about the local flow regime is made before the geometry definition in the LEVOFR routine. The physical models for the flow regimes are described in Section 16.4.1.4, and this section will describe only the decision process used to select the appropriate flow regime. The total flow regime is dependent on the local volumetric fraction of various components in the cell and on the previously established flow regime. The volumetric fraction for component i, $\alpha_{\rm i}$, is defined as follows:

$$\alpha_{\rm i} = \frac{A_{\rm i} \cdot \Delta z_{\rm i}}{A_{\rm ch,op,i} \cdot \Delta z_{\rm i}} = \frac{\theta_{\rm i}}{\theta_{\rm ch,op}} \tag{16.4-242}$$

where $A_{\rm ch,op}$ is the local cross sectional area of the open flow channel. The flow regime in each cell is determined in the following manner:

(I) If the previously established flow regime is bubbly fuel flow and

(a) if $\frac{\theta_{\rm fu,i}+\theta_{\rm se,i}}{\theta_{\rm ch,op,i}-\theta_{lu,i}} >$ CIBBLY $\cdot\, 0.7$ and

1. • if $\theta_{\rm fu,i} \geq \theta_{\rm se,i} \cdot 0.9 \rightarrow$ bubbly fuel flow regime
2. • if $\theta_{\rm fu,i} < \theta_{\rm se,i} \cdot 0.9 \rightarrow$ bubbly steel flow regime

(b) if $\frac{\theta_{\rm fu,i}+\theta_{\rm se,i}}{\theta_{\rm ch,op,i}-\theta_{lu,i}} \leq$ CIBBLY $\cdot\, 0.7$ and

1. • if $\theta_{\rm fu,i} \geq \theta_{\rm se,i} \rightarrow$ annular fuel flow regime
2. • if $\theta_{\rm fu,i} < \theta_{\rm se,i} \rightarrow$ annular steel flow regime

(II) If the previously established flow regime is bubbly steel flow and

(a) if $\frac{\theta_{\rm fu,i}+\theta_{\rm se,i}}{\theta_{\rm ch,op,i}-\theta_{lu,i}} >$ CIBBLY $\cdot\, 0.7$ and

1. • if $\theta_{\rm se,i} > \theta_{\rm fu,i} \cdot 0.9 \rightarrow$ bubbly steel flow regime
2. • if $\theta_{\rm se,i} \leq \theta_{\rm fu,i} \cdot 0.9 \rightarrow$ bubbly fuel flow regime

(b) if $\frac{\theta_{\rm fu,i}+\theta_{\rm se,i}}{\theta_{\rm ch,op,i}-\theta_{lu,i}} \leq$ CIBBLY $\cdot\, 0.7$ and

1. • if $\theta_{\rm se,i} \geq \theta_{\rm fu,i} \rightarrow$ annular steel flow regime
2. • if $\theta_{\rm se,i} < \theta_{\rm fu,i} \rightarrow$ annular fuel flow regime

(III) If the previously established flow regime is annular fuel or steel flow and

(a) if $\frac{\theta_{\rm fu,i}+\theta_{\rm se,i}}{\theta_{\rm ch,op,i}-\theta_{lu,i}} <$ CIBBLY and

1. • if $\theta_{\rm fu,i} \geq \theta_{\rm se,i} \rightarrow$ annular fuel flow regime
2. • if $\theta_{\rm fu,i} < \theta_{\rm se,i} \rightarrow$ annular steel flow regime

(b) if $\frac{\theta_{\rm fu,i}+\theta_{\rm se,i}}{\theta_{\rm ch,op,i}-\theta_{lu,i}} \geq$ CIBBLY and

1. • if $\theta_{\rm fu,i} \geq \theta_{\rm se,i} \rightarrow$ bubbly fuel flow regime
2. • if $\theta_{\rm fu,i} < \theta_{\rm se,i} \rightarrow$ bubbly steel fuel flow regime

The input constant CIBBLY defines the threshold for the transition from the annular to the bubbly flow regime. In the axial cells where the pins have not yet been disrupted the value of CIBBLY is given by the input constant CIBBIN. In disrupted nodes the value of CIBBLY is given the input constant CIBBDI. The recommended value for CIBBIN is 0.7, and for CIBBDI the recommended value is 0.2. Once a bubbly flow regime has been established, a hysteresis effect is assumed to exist and the transition threshold back to annular flow is CIBBLY $\cdot$ 0.7. Similar hysteresis effects are used for the transitions from steel to fuel and vice versa within the bubbly flow

regime. Finally, it is noted that the volume fractions used in the flow regime decision are based on the newly calculated densities and thus are consistent with the conditions at the end of the current time step.

16.4.3.9.2 Description of the Local Geometry of the Stationary and Moving Components

The definition of the local geometry of both stationary and moving components is performed by the LEGEOM routine. The physical boundaries of the rod bundle channel modeled in LEVITATE are the pin cladding and the hexcan wall. Because of the different behavior of these boundaries, the channel is visualized as being divided into two separate channels, one associated with the fuel pins and another associated with the hexcan wall, as shown in Figure 16.4.16. The flow area is partitioned between these two channels in proportion to the wetted perimeter of the boundaries:

$$A_{\text{ch,cl,i}} = \frac{A_{\text{ch,i}}}{L_{\text{cl,i}} + L_{\text{sr,i}}} \cdot L_{\text{cl,i}} \tag{16.4-243}$$

$$A_{\text{ch,sr,i}} = \frac{A_{\text{ch,i}}}{L_{\text{cl,i}} + L_{\text{sr,i}}} \cdot L_{\text{sr,i}} \tag{16.4-244}$$

where $L_{\text{cl,i}}$ and $L_{\text{sr,i}}$ represent the perimeter of the pins and hexcan wall in cell i, respectively. It is emphasized that LEVITATE models the whole subassembly, and not one pin representing the subassembly. Thus, $L_{\text{cl,i}}$ represents the perimeter of all the pins in the bundle. In the special situation when no pins are present at a certain axial location due to pin disruption, the perimeter $L_{\text{cl,i}} = 0$, and the area of the channel associated with the cladding, $A_{\text{ch,cl,i}}$ becomes zero. Each of the two channel is treated, for the purpose of defining the geometry, as a rectangular channel, with one dimension being the perimeter L and the other a characteristic length, l. This characteristic length is defined as follows:

$$l_{sr,i} = \frac{A_{\text{ch,sr,i}}}{L_{\text{sr,i}}} \tag{16.4-245}$$

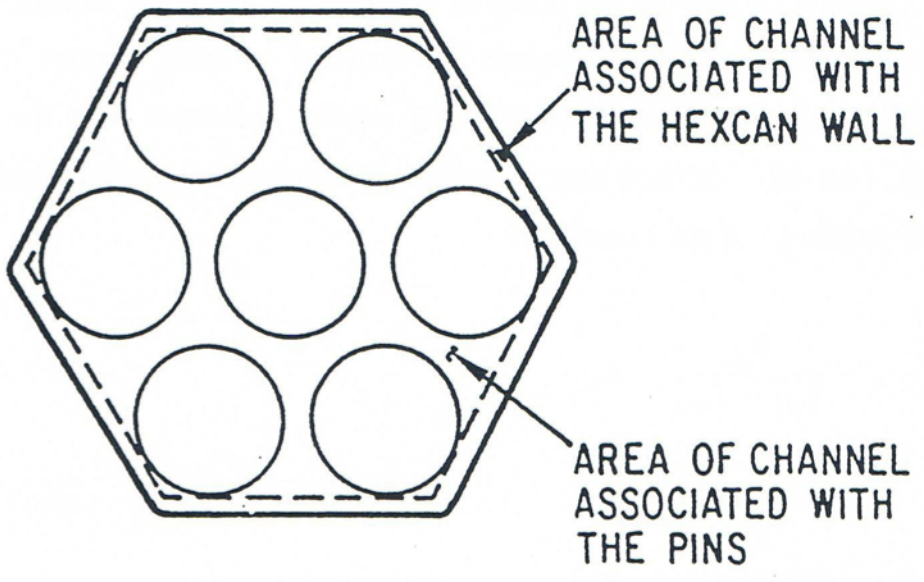


Figure 16.4.16: Partition of Subassembly Flow Area into Two Distinct Channels

The characteristics length l characterizes the width of the channel and serves as an indicator for the maximum thickness of the fuel crusts that can form on the pins and/or the hexcan wall. The conceptual representation of

these channels is presented in Figure 16.4.17. The fuel crust, when present, is characterized by length $L_{\text{ff,cl,i}}$ (and $L_{\text{ff,sv,i}}$) and thickness $l_{\text{ff,cl,i}}$ (and $l_{\text{ff,sv,i}}$). However, instead of storing the crust L_{ff}, two area coefficient are used:

$$C_{\text{ff,cl,i}} = \frac{L_{\text{ff,cl,i}}}{L_{\text{cl,i}}} \tag{16.4-246}$$

and

$$C_{\text{ff,sr,i}} = \frac{L_{\text{ff,sr,i}}}{L_{\text{sr,i}}} \tag{16.4-247}$$

As explained in Section 16.5, the fuel crust can occasionally contain molten or frozen steel, which is taken into account when the crust size is calculated. The crust growth is dependent on the initial conditions, the amount of new frozen fuel, and the local flow regime. This process will be presented later in this chapter.

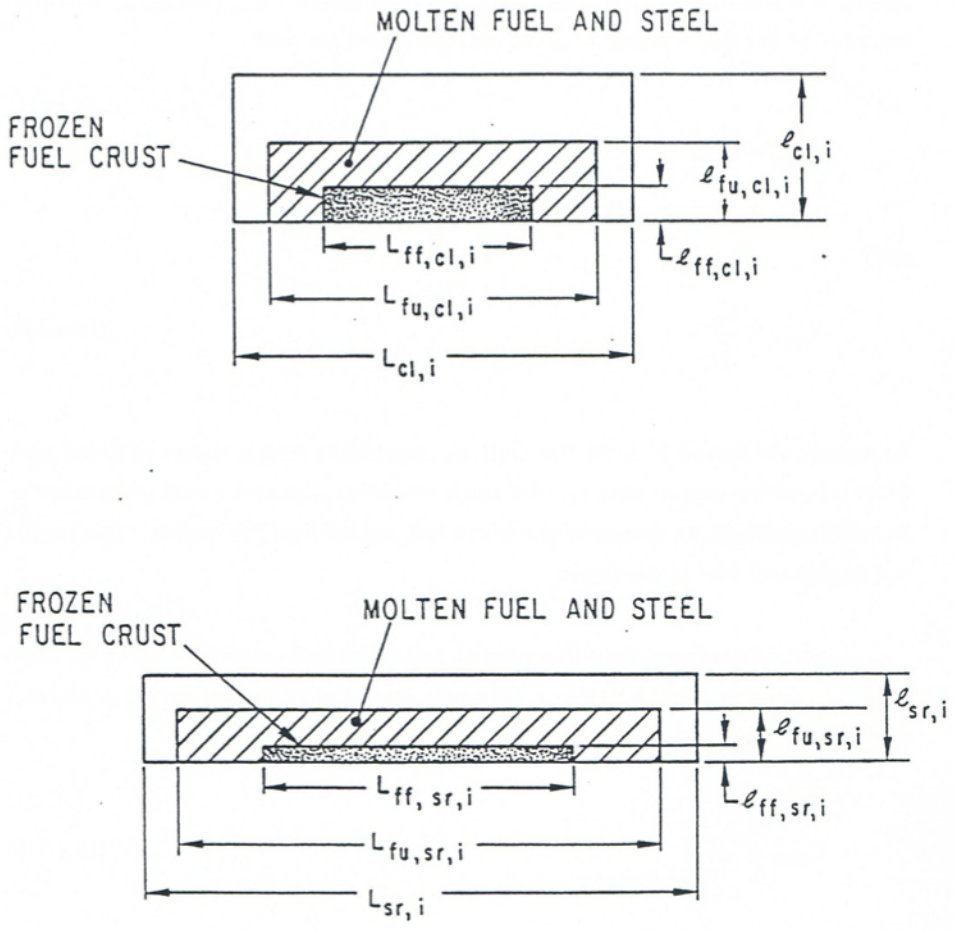


Figure 16.4.17: Conceptual Representation of the Pin and Hexcan Wall Channels

The configuration of the molten material, fuel and/or steel, is characterized by the length $L_{\text{fu,cl,i}}$ ($L_{\text{fu,sr,i}}$) and thickness $l_{\text{fu,cl,i}}$ ($l_{\text{fu,sr,i}}$). However, instead of storing the length L_{fu}, two are coefficients are used:

$$C_{\text{fu,cl,i}} = \frac{L_{\text{fu,cl,i}}}{L_{\text{cl,i}}} \tag{16.4-248}$$

$$C_{\text{ff,sr,i}} = \frac{L_{\text{ff,sr,i}}}{L_{\text{sr,i}}} \tag{16.4-249}$$

Note that the coefficient $C_{\text{fu,cl,i}}$ represents the fraction of clading area covered by molten fuel and fuel crust. The same applies to $C_{\text{fu,sr,i}}$. All the contact areas required for the energy and momentum equations are defined by using the lateral area of the clad $A^L_{\text{cl,i}}$ and hexcan wall $A^L_{\text{sr,i}}$ and the appropriate area coefficients. Other area conditions used are:

$C_{\text{fu,f}l\text{,cl,i}}$ - fraction of cladding crust area covered by molten fuel in cell i

$C_{\text{fu,ff,sr,i}}$ - fraction of structure crust area covered by molten fuel in cell i

$C_{\text{se,cl,i}}$ - fraction of cladding area covered by molten steel and fuel crust in cell i

$C_{\text{se,sr,i}}$ - fraction of hexcan wall area covered by molten steel and fuel crust in cell i

$C_{\text{se,ff,cl,i}}$ - fraction of cladding crust area covered by molten steel in cell i

$C_{\text{se,ff,sr,i}}$ - fraction of hexcan wall crust area covered by molten steel in cell i

To each of these coefficients is attributed a specific value in the routine LEGEOM, depending on the flow regime and the initial conditions. Most of these values are self-explanatory and can be understood by looking a Figure 16.4.1, which illustrates the material configuration in each LEVITATE flow regime. Some additional comments are required for the partial annual fuel flow regime. To fully describe this flow regime, the following assumptions were made:

1. The molten fuel film maintains a constant ratio $\frac{l_{\text{fu,i}}}{L_{\text{fu,i}}}$ both for the clad and the hexcan wall. This ratio is defined by:

$$\frac{l_{\text{fu,i}}}{L_{\text{fu,i}}} = \frac{A_{\text{ch,op,i}} \cdot \text{CIANLR}}{\left(L_{\text{cl,i}} + L_{\text{sr,i}}\right)^2} \tag{16.4-250}$$

when no fuel crusts are present. It is built into the Eq. (16.4-252) and the user can affect it value only by changing the input constant CIANLR. This assumption leads to a gradual increase of the film-covered perimeter, together with a film thickness increase, whenever the amount of molten fuel increases.

1. The partial annual flow becomes fully annular when the volume fraction of the molten fuel/steel reaches a certain input value, i.e., when

$$\frac{\theta_{\text{fu,i}} + \theta_{\text{sc,i}}}{\theta_{\text{ch,op,i}}} = \text{CIANLR} \tag{16.4-251}$$

2. The existing fuel crust should be taken into account when calculating the coefficients $C_{\text{fu,cl,i}}$ and $C_{\text{fu,sr,i}}$.

These assumptions lead to the following definition for the area coefficient $C_{\text{fu,cl,i}}$ in the annular fuel flow regime

$$C_{\text{fu,cl,i}} = \begin{cases} \frac{\theta_{\text{fu,i}}+\theta_{\text{sc,i}}}{\theta_{\text{ch,op}}} + \frac{A_{\text{ff,cl,i}}}{A_{\text{ch,cl,i}}} / \text{CIANLR} \\ 1, \qquad \text{if the above expression} > 1 \end{cases} \tag{16.4-252}$$

For nodes where the fuel pins are undisrupted, CIANLR is set equal to the input value CIANIN, currently 0.5. For disrupted nodes, CIANLR is equal CIANDI. A similar formula applies to $C_{\text{fu,sr,i}}$.

Another aspect treated in LEGEOM, which will be discussed in this section, is the change in the fuel crust geometry due to additional fuel freezing. It is noted that other aspects such as steel freezing, crust breakup and remelting are treated in the routine LEFREZ and will be discussed in Section 16.5. Also, other aspects related to the chunk geometry are treated in the LETRAN routine and will be introduced in Section 16.4.3.10.

The amounts of fuel/steel which have to be added to the fuel crusts due to freezing are calculated in the LEFREZ routine in the previous time step. In the routine LEGEOM, these amounts are converted to corresponding are changes $\Delta A_{\text{ff,cl,i}}$ and $\Delta A_{\text{ff,se,i}}$. The decision on how to modify the crust parameters $L_{\text{ff,cli,i}}$, $l_{\text{ff,cl,i}}$, $L_{\text{ff,sr,i}}$ and $l_{\text{ff,sr,i}}$ is made in the following way (we will refer to the crust on the cladding only):

If no previous fuel crest is present and if the local fuel flow regime is partially annular flow, the original crust has the same thickness as the molten fuel film; however, if the local fuel flow regime is bubbly, the original crust is rather thin and the area coefficient $C_{\text{ff,cl,i}}$ is obtained as follows:

$$C_{\text{ff,cl,i}} = \begin{cases} \sqrt{\frac{A_{\text{ff,cl,i}}}{A_{\text{ch,cl,i}}}/\text{CIBBFZ}} \\ 1, \qquad \text{if above expression} > 1 \end{cases} \tag{16.4-253}$$

where CIBBFZ is a constant, which is currently set equal to the input constant CIBBDI. If a previously formed crust is present the thickness of the crust is first increased until it reaches the thickness of he fuel film or the channel characteristic size $l_{\text{cl,i}}$, whichever comes first. Then the length of the crust $L_{\text{ff,cl,i}}$ is increased until $C_{\text{ff,cl,i}} = 1$.

16.4.3.9.3 Description of the Fuel/Steel Chunk Geometry

The chunks modeled in LEVITATE have a cylindrical geometry, as shown in Figure 16.4.18. They are characterized by the radius R, which is different in each axial cell. The length is related to the radius by the input constant ASRALU, which defines the ratio $L/2R$ and has currently the recommended value 1. Each chunk can contain either fuel, or steel or both. The density of the steel chunks is constant and determined by the input variable RHSESO. The fuel chunks, however, have a variable density, which accounts for the possible porosity of the frozen fuel.

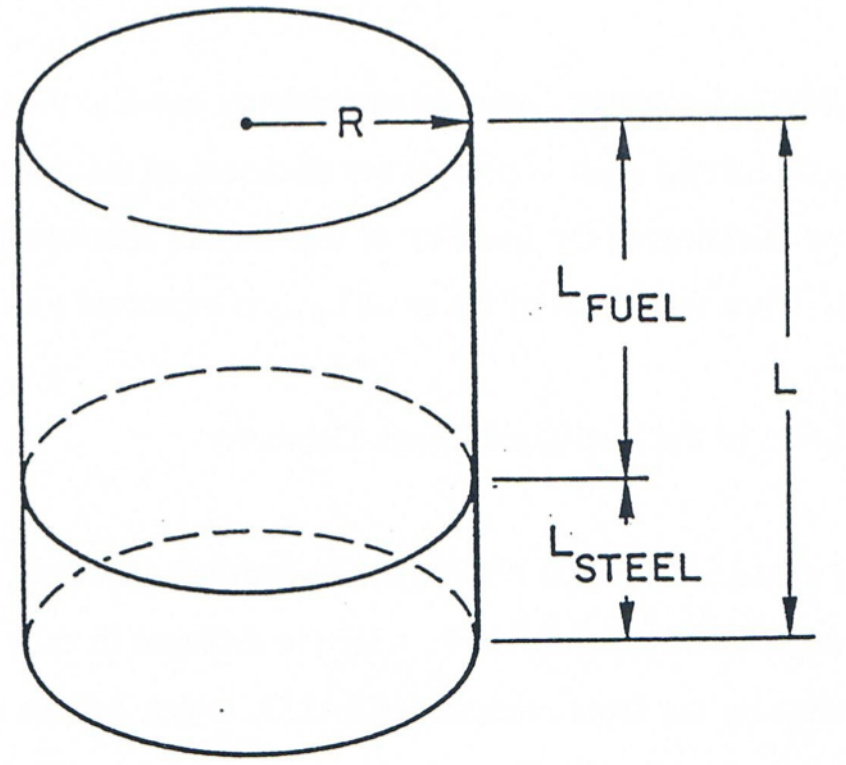


Figure 16.4.18: Fuel/Steel Chunk Geometry

The original geometry of the chunks depends on their origin. Several mechanisms have been identified which lead to the formation of solid fuel/steel chunks:

A. Disruption of the original fuel pins, which can occur when the cladding is locally molten, leading to the formation of relatively large fuel chunks. In this case the characteristic radius of the chunks is determined from:

$$R_{\text{fu,i}}^{\text{NEW}} = \sqrt{\Delta R_{\text{pin,i}} \cdot \Delta L_{\text{perimeter,pin,i}}} \tag{16.4-254}$$

where

$\Delta R_{\text{pin,i}}$ - is the thickness of the solid fuel pin wall which separate the pin cavity from the coolant channel

$\Delta L_{\text{perimeter,pin,i}}$ - is the characteristic size of the chunks which is obtained by dividing the pin perimeter by the number of radial cracks present at any axial location. The number of radial crack currently used is 8.

If the input variable RALUDI is zero, the code will use the above formula to determine the radius of the new chunks. However, this calculation can be overridden by setting RALUDI to a non-zero value. The radius of the newly formed chunks will be set equal to RALUDI.

B. Breakup of the frozen fuel crust that was formed previously on the cladding and hexcan wall. The characteristic radius of these chunks is determined from:

$$R_{lu,\text{i}}^{\text{NEW}} = \sqrt{\Delta A_{\text{ffc,i}}} \tag{16.4-255}$$

or

$$R_{lu,\text{i}}^{\text{NEW}} = \sqrt{\Delta A_{\text{ffs,i}}} \tag{16.4-256}$$

where

$\Delta A_{\text{ffc,i}}$ - is the cross sectional area of the frozen fuel crust which breaks up during the current time step, in the axial cell i

$\Delta A_{\text{ffs,I}}$ - is similar to $\Delta A_{\text{ffc,i}}$, but applies to the fuel crust associated with the hexcan wall.

C. Local bulk freezing of the fuel when no solid support for crust formation is present. The characteristic radius of the chunks is obtained from:

$$R_{\text{fu,i}}^{\text{NEW}} = \sqrt{\Delta A_{\text{fu,i}}} \tag{16.4-257}$$

where

$\Delta A_{\text{fu,I}}$ - is the change in the cross sectional area of the molten fuel component due to removal of frozen fuel.

The volume of all the newly formed chunks in the axial cell i is calculated as follows:

$$\Delta V_{lu,\text{i}} = \left(\frac{\Delta \rho'_{fl,\text{i}}}{\rho_{\text{fl,i}}} + \frac{\Delta \rho'_{sl,\text{i}}}{\rho_{sl,\text{i}}} \right) \cdot \Delta z_{\text{i}} \cdot \text{AXMX} \tag{16.4-258}$$

where

$\Delta \rho'_{\text{fl,i}}$ - is the generalized density of the fuel component in the newly formed chunks

$\Delta \rho'_{\text{sl,i}}$ - is the generalized density of the fuel component in the newly formed chunks

The number of new chunks is obtained by dividing the total volume $\Delta V_{lu,\text{i}}$ by the volume of a single chunk:

$$\Delta N_{\text{fu,i}} = \frac{\Delta V_{\text{fu,i}} \cdot \text{ASRALU}}{2 \cdot \pi \cdot \left(R_{\text{lu,i}}^{\text{NEW}}\right)^3} \tag{16.4-259}$$

The new chunks will generally have different characteristics from the chunks already present in a cell. Because only one type of chunks can exist in one cell, the two categories are merged in one chunk population with properties reflecting the characteristics of both original components. This process is outlined below. Note that

the same problem can arise in any cell even without the formation of new chunks due to the flux of chunk from neighboring cells. The first step in the merging of the two chunk categories is to define the radius of the resulting chunks.

$$R_{l\mathrm{u,i}}^{n+1} = \frac{R_{l\mathrm{u,i}} \cdot \left[N_{l\mathrm{u,i}} \cdot R_{\mathrm{lu,i}}^{\mathrm{ILUMER}}\right] + R_{l\mathrm{u,i}}^{\mathrm{NEW}} \cdot \left[\Delta N_{l\mathrm{u,i}} \cdot \left(R_{l\ \mathrm{u,i}}^{\mathrm{NEW}}\right)^{\mathrm{ILUMBER}}\right]}{N_{l\mathrm{u,i}} \cdot R_{l\mathrm{u,i}}^{\mathrm{ILUMER}} + \Delta N_{l\mathrm{u,i}} \cdot \left(R_{l\mathrm{u,i}}^{\mathrm{NEW}}\right)^{\mathrm{ILUMBER}}} \tag{16.4-260}$$

The new chunk radius is a weighted average of the radii of the two chunk categories. The weights can be selected b the user, by changing the input integer ILUMER, which can have the values 0, 1, 2, or 3. For ILUMER $= 0$, the weights are equal to the number of chunks in the two populations. The recommended value is 3, for which the weights are proportional to the total volume. The number of resulting chunks is given by:

$$N_{l\mathrm{u,i}} = \frac{V_{l\mathrm{u,i}} + \Delta V_{l\mathrm{u,i}}}{2\pi \left(R_{l\mathrm{u,i}}^{n+1}\right)^3} \mathrm{ASRALU} \tag{16.4-261}$$

where the radius $R_{l,\mathrm{u,i}}^{n+1}$ is defined by 16.4-224-7. Finally the fraction of steel in the chunks is given by:

$$\begin{aligned} \mathrm{FR}_{sl,l\mathrm{u,i}}^{n+1} &= \frac{L_{\mathrm{STEEL},l\mathrm{u,i}}}{L_{\mathrm{STEEL},l\mathrm{u,i}} + L_{\mathrm{FUEL},l\mathrm{u,i}}} \\ &= \frac{\left(\rho'_{sl,\mathrm{i}} + \Delta\rho'_{sl,\mathrm{i}}\right) \mathrm{AXMX}\Delta z_{\mathrm{i}}}{2\pi \mathrm{ASRALU} N_{l\mathrm{u,i}} R_{l\mathrm{u,i}}^{n+1} \rho_{sl}} \end{aligned} \tag{16.4-262}$$

The quantities L_{FUEL} and L_{STEEL} are introduced in Figure 16.4.18.

If the input variable RALUFZ is zero, then the code will use the above formulas to determine the radius of the new chunks formed by crust break-up and/or bulk freezing. However, if RALUFZ is not zero, the radius of the newly formed chunks will be set equal to RALUFZ.

16.4.3.10 Source Terms in the Energy and Momentum Equations

This section completes the description of the channel hydrodynamic model. It presents in detail the formulation of the source terms used in the energy equations, as introduced previously. It also presents some components of the source terms used in the momentum equations. The description of these source terms has been delayed until now for two reasons: first, the geometry elements introduced in Section 16.4.3.9 are needed for the definition of the source terms; second, all the elements described below are calculated in the routine LETRAN, and thus it was deemed preferable to describe them separately from the energy and momentum equations which are solved elsewhere.'

16.4.3.10.1 Source Terms in the Energy Equations

In the energy conservation equations, we introduced the source terms in the following form:

(1) For the fuel energy Eq. (16.4-16):

$$\begin{aligned} \sum_{\mathrm{k}} H'_{\mathrm{fu,k,i}} \cdot \Delta T_{\mathrm{fu,k,i}} = {} & H'_{\mathrm{fu,Na,i}} \cdot \Delta T_{\mathrm{fu,Na,i}} + H'_{\mathrm{fu,cl,i}} \cdot \Delta T_{\mathrm{fu,cl,i}} \\ & + H'_{\mathrm{fu},fl,\mathrm{i}} \cdot \Delta T_{\mathrm{fu},fl,\mathrm{i}} + H'_{\mathrm{fu},sl,\mathrm{i}} \cdot \Delta T_{\mathrm{fu},sl,\mathrm{i}} \\ & + H'_{\mathrm{fu,sr,i}} \cdot \Delta T_{\mathrm{fu,sr,i}} + H'_{\mathrm{se,fu,i}} \cdot \Delta T_{\mathrm{fu,se,i}} \\ & + H'_{\mathrm{fu,ffc,i}} \cdot \Delta T_{\mathrm{fu,ffc,i}} + H'_{\mathrm{fu,ffs,i}} \cdot \Delta T_{\mathrm{fu,ffs,i}} \end{aligned} \tag{16.4-263}$$

(1) For the steel energy Eq. (16.4-28):

$$\sum_{\mathrm{k}} H'_{\mathrm{se,k,i}} \cdot \Delta T_{\mathrm{se,k,i}} = H'_{\mathrm{se,fu,i}} \cdot \Delta T_{\mathrm{fu,se,i}} + H'_{\mathrm{se,cl,i}} \cdot \Delta T_{\mathrm{se,cl,i}} + H'_{\mathrm{se,sr,i}} \cdot \Delta T_{\mathrm{se,sr,i}} + H'_{\mathrm{se},fl,\mathrm{i}} \cdot \Delta T_{\mathrm{se},fl,\mathrm{i}} + H'_{\mathrm{se},sl,\mathrm{i}} \cdot \Delta T_{\mathrm{se},sl,\mathrm{i}} + H'_{\mathrm{se,Na,i}} \cdot \Delta T_{\mathrm{se,Na,i}} + H'_{\mathrm{se,ffc,i}} \cdot \Delta T_{\mathrm{se,ffc,i}} + H'_{\mathrm{se,ffs,i}} \cdot \Delta T_{\mathrm{se,ffs,i}} \tag{16.4-264}$$

(1) For the fuel chunk energy

$$\sum_{\mathrm{k}} H'_{\mathrm{f}l,\mathrm{k,i}} \cdot \Delta T_{\mathrm{f}l,\mathrm{k,i}} = H'_{\mathrm{fu},fl,\mathrm{i}} \cdot \Delta T_{\mathrm{f}l,\mathrm{fu,i}} + H'_{\mathrm{se},fl,\mathrm{i}} \cdot \Delta T_{\mathrm{f}l,\mathrm{se,i}} + H'_{\mathrm{Na},fl,\mathrm{i}} \cdot \Delta T_{\mathrm{f}l,\mathrm{Na,i}} + H'_{\mathrm{cl},fl,\mathrm{i}} \cdot \Delta T_{\mathrm{f}l,\mathrm{cl,i}} + H'_{\mathrm{sr},fl,\mathrm{i}} \cdot \Delta T_{\mathrm{f}l,\mathrm{sr,i}} + H'_{\mathrm{ffc},fl,\mathrm{i}} \cdot \Delta T_{\mathrm{f}l,\mathrm{ffc,i}} + H'_{\mathrm{ffs},fl,\mathrm{i}} \cdot \Delta T_{\mathrm{f}l,\mathrm{ffs,i}} + H'_{\mathrm{f}l,sl,\mathrm{i}} \cdot \Delta T_{\mathrm{f}l,sl,\mathrm{i}} \tag{16.4-265}$$

(1) For the steel chunk energy equation:

$$\sum_{\mathrm{k}} H'_{sl,\mathrm{k,i}} \cdot \Delta T_{sl,\mathrm{k,i}} = H'_{\mathrm{fu},sl,\mathrm{i}} \cdot \Delta T_{sl,\mathrm{fu,i}} + H'_{\mathrm{se},sl,\mathrm{i}} \cdot \Delta T_{sl,\mathrm{se,i}} + H'_{\mathrm{Na},sl,\mathrm{i}} \cdot \Delta T_{sl,\mathrm{Na,i}} + H'_{\mathrm{cl},sl,\mathrm{i}} \cdot \Delta T_{sl,\mathrm{cl,i}} + H'_{\mathrm{sr},sl,\mathrm{i}} \cdot \Delta T_{sl,\mathrm{sr,i}} + H'_{\mathrm{ffc},sl,\mathrm{i}} \cdot \Delta T_{sl,\mathrm{ffc,i}} + H'_{\mathrm{ffs},sl,\mathrm{i}} \cdot \Delta T_{sl,\mathrm{ffs,i}} + H'_{\mathrm{f}l,sl,\mathrm{i}} \cdot \Delta T_{sl,\mathrm{f}l,\mathrm{i}} \tag{16.4-266}$$

(1) For the gas-mixture equation

$$\sum_{\mathrm{k}} H'_{\mathrm{k,Mi,i}} \cdot \Delta T_{\mathrm{k,Mi,i}} = H'_{\mathrm{Na},cl,\mathrm{i}} \cdot \Delta T_{\mathrm{cl,Na,i}} + H'_{\mathrm{se,Na,i}} \cdot \Delta T_{\mathrm{se,Na,i}} + H'_{\mathrm{Na,sr,i}} \cdot \Delta T_{\mathrm{sr,Na,i}} + H'_{\mathrm{fu,Na,i}} \cdot \Delta T_{\mathrm{fu,Na,i}} + H'_{\mathrm{Na},fl,\mathrm{i}} \cdot \Delta T_{fl,\mathrm{Na,i}} + H'_{\mathrm{Na},sl,\mathrm{i}} \cdot \Delta T_{sl,\mathrm{Na,i}} + H'_{\mathrm{ffc,Na,i}} \cdot \Delta T_{\mathrm{ffc,Na,i}} + H'_{\mathrm{ffs,Na,i}} \cdot \Delta T_{\mathrm{ffs,Na,i}} \tag{16.4-267}$$

where

$$H'_{\mathrm{j,k,i}} = H_{\mathrm{j,k,i}} \cdot \frac{A^L_{\mathrm{j,k,i}}}{\Delta z_{\mathrm{i}} \cdot \mathrm{AXMX}} \tag{16.4-268}$$

We will now proceed to define all generalized heat-transfer coefficients that appear in Eq. (16.4-263) through Eq. (16.4-268). As shown in Eq. (16.4-268) both $H_{\mathrm{j,k,i}}$ and $A_{\mathrm{j,k,i}}$ will have to be defined. Special reference will be made to each flow regime, and a short description of the physical model will be included.

16.4.3.10.1.1 Heat-transfer Coefficient between Fuel and Sodium

For the annular and bubbly steel flow regimes, there is no contact between the fuel and sodium. Therefore, the heat-transfer coefficient is zero:

$$H'_{\mathrm{fu,Na,i}} = 0 \tag{16.4-269}$$

For the annular fuel flow regime, the heat-transfer coefficient is defined as follows:

$$H_{\mathrm{fu,Na,i}} = \frac{1}{\frac{1}{H_{\mathrm{fu,i}}} + \frac{1}{H_{\mathrm{Na,i}}}} \tag{16.4-270}$$

where $H_{\mathrm{fu,i}}$ is a convection-type transfer coefficient on the fuel side:

$$H_{\mathrm{fu,i}} = \frac{1}{D_{\mathrm{H,fu,i}}} \cdot \mathrm{CIA3} \cdot \mu_{\mathrm{fu}} \cdot c_{\mathrm{p,fu}} \cdot \mathrm{Re}^{0.8}_{\mathrm{fu,i}} \tag{16.4-271}$$

and the sodium heat-transfer coefficient in the two-phase region is defined as:

$$H_{\text{Na,i}} = \begin{cases} 10^5, & \text{if } \alpha_{\text{Na,i}} < 0.5 \\ \left[10^5 \cdot (1-\alpha_{\text{na,i}}) + H_{\text{Na,i}}^{\text{Single Phase}} \cdot (\alpha_{\text{Na,i}} - 0.5)\right] \cdot 2, & \text{if } 0.5 \le \alpha_{\text{Na,i}} < 1 \end{cases} \tag{16.4-272}$$

where $H_{\text{Na,i}}^{\text{Single,Phase}}$ is defined in Eq. (16.4-273).

The convective sodium heat-transfer coefficient in the single-phase region is defined below:

$$H_{\text{Na,i}}^{\text{Single Phase}} = \Bigg\{ \text{C1} \cdot \left[D_{\text{H,Mi,i}} \cdot \rho_{\text{Mi,i}} \cdot |u_{\text{Mi,i}} - u_{\text{fu,i}}| \cdot C_{\text{p,Mi,i}} \cdot \frac{1}{k_{\text{Mi,i}}} \right]^{\text{C2}} + \text{C3} \Big\} \cdot \frac{k_{\text{Mi,i}}}{D_{\text{H,Mi,i}}} \text{ for } \alpha_{\text{Na,i}} > 1 \tag{16.4-273}$$

The area of contact between sodium and molten fuel is defined as follows:

$$\begin{aligned} A_{\text{fu,Na,i}}^{L} = {} & A_{\text{cl,i}}^{L} \cdot [(C_{\text{fu,cl,i}} - C_{\text{ff,cli,i}}) + C_{\text{ff,cl,i}} \cdot C_{\text{fu,ff,cl,i}}] \\ & + A_{\text{sr,i}}^{L} \cdot [(C_{\text{fu,sr,i}} - C_{\text{ff,sr,i}}) + C_{\text{ff,sr,i}} \cdot C_{\text{fu,ff,sr,i}}] \end{aligned} \tag{16.4-274}$$

For the bubbly fuel flow regime, the heat-transfer between sodium and fuel is described as taking place between a continuous fuel component and spherical sodium bubbles. The number of sodium bubbles (and thus their radius) varies from one bubble at the transition boundary between annular and bubbly flow to a maximum number determined by a maximum number determined by a minimum radius of the bubble:

$$R_{\text{min,bubble,i}} = 0.05 R_{\text{max,bubble,i}} \tag{16.4-275}$$

$$R_{\text{max,bubble,i}} = \begin{cases} R_1 = \left(\frac{3 \cdot V_{\text{Na,i}}}{4 \cdot \pi}\right)^{1/3} & \text{if } R_1 < D_{\text{H,Mi,i}} \cdot 0.5 \\ D_{\text{H,Mi,i}} \cdot 0.5 & \text{if } R_1 \ge D_{\text{H,Mi,i}} \cdot 0.5 \end{cases} \tag{16.4-276}$$

The actual radius of the bubbles is obtained by assuming an exponential variation between R_{max} and R_{min}, dependent on how far from the flow regime transition we are in the cell i:

$$\begin{aligned} R_{\text{bubble,i}} = {} & R_{\text{min,bubble,i}} + (R_{\text{max,bubble,i}} - R_{\text{min,bubble,i}}) \\ & \cdot \exp\left[\frac{-(\theta_{\text{Mi,bubbly}} - \theta_{\text{Mi,i}})}{\theta_{\text{Mi,i}}}\right] \end{aligned} \tag{16.4-277}$$

where $\theta_{\text{Mi,bubbly}}$ is the mixture volume fraction required for the transition from the annular to bubbly flow.

The number of bubbles is defined by:

$$N_{\text{bubble,i}} = \frac{3 \cdot V_{\text{Na,i}}}{4 \cdot \pi \cdot R_{\text{bubbles,i}}^3} \tag{16.4-278}$$

and the heat-transfer area becomes:

$$A_{\text{fu,Na,i}}^{L} = N_{\text{bubble,i}} \cdot 4\pi \cdot R_{\text{bubble,i}}^2 \tag{16.4-279}$$

The heat-transfer coefficient is determined by the heat-transfer resistance on the fuel side and the resistance of the bubble:

$$H_{\text{fu,Na,i}} = \frac{1}{\frac{1}{H_{\text{fu,i}}} + \frac{1}{H_{\text{Na,i}}}} \tag{16.4-280}$$

where $H_{\text{fu,i}} = 3 \cdot 10^4$ J/m^2 K-s and wad determined by analyzing the transient solution describing the propagation of a step change in temperature in a semi-infinite medium. For a typical LEVITATE time step Δt and a temperature change ΔT, $H_{\text{fu,i}}$ was determined from:

$$H_{\text{fu,i}} = \frac{1}{\Delta T \cdot \Delta t} \int_0^{\Delta t} Q\,(x=0)\,\text{dt} \tag{16.4-281}$$

Where $Q\,(x=0)$ is the heat flux at the boundary of the semi-infinite medium, in J/m^2·s.

The heat-transfer coefficient in the bubble, $H_{\text{Na,i}}$ is defined by Eq. (16.4-272) for the two-phase sodium region. For single-phase sodium:

$$H_{\text{Na,i}} = \frac{k_{\text{Mi,i}}}{R_{\text{bubble,i}}} \quad \text{if } \alpha_{\text{Na,i}} = 1 \tag{16.4-282}$$

16.4.3.10.1.2 Heat-transfer Coefficient between Fuel and Cladding or Fuel and Structure

For the annular and bubbly steel flow regimes the fuel is in contact with neither the cladding nor the structure. Therefore, the heat transfer coefficients are zero:

$$H'_{\text{fu,cl,i}} = H'_{\text{fu,sr,i}} = 0 \tag{16.4-283}$$

For the annular fuel flow regime, the heat-transfer coefficient from fuel to cladding and structure is defined as:

$$H_{\text{fu,cl,i}} = H_{\text{fu,sr,i}} = \frac{1}{D_{\text{H,fu,i}}} \cdot \text{CIA3} \cdot \mu_{\text{fu,i}} \cdot C_{\text{p,fu}} \cdot \text{Re}_{\text{fu,i}}^{0.8} \tag{16.4-284}$$

and the respective heat-transfer areas are:

$$\begin{aligned} A^L_{\text{fu,cl,i}} &= A^L_{\text{cl,i}} \cdot (C_{\text{fu,cl,i}} - C_{\text{ff,cl,i}}) \\ A^L_{\text{fu,sr,i}} &= A^L_{\text{sr,i}} \cdot (C_{\text{fu,sr,i}} - C_{\text{ff,sr,i}}) \end{aligned} \tag{16.4-285}$$

For the bubbly fuel flow regime, the heat-transfer coefficients are defined as follows:

$$H_{\text{fu,cl,i}} = H_{\text{fu,sr,i}} = \frac{1}{D_{\text{H,fu,i}}} \cdot \text{CIA3} \cdot \mu_{\text{fu}} \cdot C_{\text{p,fu}} \cdot \text{Re}_{\text{fu,i}}^{0.8} + \frac{4 \cdot k_{\text{fu}}}{D_{\text{H,fu,i}}} \tag{16.4-286}$$

and the respective heat-transfer areas are:

$$A^L_{\text{fu,cl,i}} = A^L_{\text{cl,i}}; \quad A^L_{\text{fu,sr,i}} = A^L_{\text{sr,i}} \tag{16.4-287}$$

16.4.3.10.1.3 Heat Transfer between Fuel and Frozen Fuel Crusts on Cladding and Structure

The heat transfer coefficients between the fuel and frozen fuel crusts on either cladding or structure are zero in the annular and bubbly steel flow regimes:

$$H'_{\text{fu,ffc,i}} = H'_{\text{fu,ffs,i}} = 0 \tag{16.4-288}$$

For the annual fuel flow regime, the heat-transfer coefficients are defined as:

$$H_{\text{fu,ffc,i}} = \frac{1}{\frac{1}{H_{\text{fu,i}}} + \frac{0.5 \cdot l_{\text{ff,cl,i}}}{k_{\text{fu}}}} \tag{16.4-289}$$

$$H_{\text{fu,ffs,i}} = \frac{1}{\frac{1}{H_{\text{fu,i}}} + \frac{0.5 \cdot l_{\text{ff,sr,i}}}{k_{\text{fu}}}} \tag{16.4-290}$$

where $H_{\text{fu,i}}$ is given by Eq. (16.4-284) and $l_{\text{ff,cl,i}}$ and $l_{\text{ff,sr,i}}$ have been defined in Section 16.4.3.9, which describes the local geometry. The respective heat-transfer areas are:

$$A_{\text{fu,ffc,i}} = A^L_{\text{cl,i}} \cdot C_{\text{ff,cl,i}} \cdot C_{\text{fu,ff,cl,i}} \tag{16.4-291}$$

and

$$A_{\text{fu,ffs,i}} = A^L_{\text{sr,i}} \cdot C_{\text{ff,sr,i}} \cdot C_{\text{fu,ff,sr,i}} \tag{16.4-292}$$

For the bubbly fuel flow regime, the heat-transfer coefficients are given by Eq. (16.4-289) and Eq. (16.4-290), but $H_{\text{fu,i}}$ in these is given by Eq. (16.4-286) rather than Eq. (16.4-284). The heat-transfer areas are given by:

$$A_{\text{fu,ffc,i}} = A^L_{\text{cl,i}} \cdot C_{\text{ff,cl,i}} \tag{16.4-293}$$

$$A_{\text{fu,ffs,i}} = A^L_{\text{sr,i}} \cdot C_{\text{ff,sr,i}} \tag{16.4-294}$$

16.4.3.10.1.4 Heat-transfer Coefficient between Fuel and Steel

In the annular and bubbly steel flow regimes, the fuel exists in the form of droplets imbedded in the continuous molten steel. The radius of the fuel droplets varies between $R_{\text{fu,droplet,min,i}} = 1 \cdot 10^{-4}$ m and $R_{\text{fu,droplet,max,i}} = l_{\text{fu,cl,i}}$. The actual radius is calculated as follows:

If $1 \geq \frac{\theta_{\text{fu,i}}}{\theta_{\text{se,i}}} \geq 0.5$

$$R_{\text{fu,droplet,i}} = \left[R_{\text{fu,droplet,min,i}} \cdot \left(1 - \frac{\theta_{\text{fu,i}}}{\theta_{\text{se,i}}} \right) + R_{\text{fu,droplet,max,i}} \cdot \left(\frac{\theta_{\text{fu,i}}}{\theta_{\text{se,i}}} - 0.5 \right) \right] \cdot 2 \tag{16.4-295}$$

If $\frac{\theta_{\text{fu,i}}}{\theta_{\text{se,i}}} < 0.5$, then

$$R_{\text{fu,droplet,i}} = R_{\text{fu,droplet,min,i}} \tag{16.4-296}$$

The number of fuel droplets and heat-transfer area are calculated from:

$$N_{\text{fu,droplets,i}} = \frac{3V_{\text{fu,i}}}{4\pi \cdot R^3_{\text{fu,droplet,i}}} \tag{16.4-297}$$

and

$$A^L_{\text{se,fu,i}} = N_{\text{fu,droplets,i}} \cdot 4\pi \cdot R^3_{\text{fu,droplet,i}} \tag{16.4-298}$$

The heat-transfer coefficient is obtained as follows:

$$H_{\text{se,fu,i}} = \frac{1}{\frac{R_{\text{fu,droplet,i}}}{k_{\text{fu}}} + \frac{1}{H_{\text{se,i}}}} \tag{16.4-299}$$

where $H_{\text{se,i}} = 4 \cdot 10^4$ and is obtained in a manner similar to Eq. (16.4-281).

For the annual and bubbly fuel flow regimes, the situation is reversed, and the steel exists in the form of droplets imbedded in the continuous molten fuel. The radius of the steel droplet is defined as follows:

For $\frac{\theta_{\text{se,i}}}{\theta_{\text{fu,i}}} > 0.5$

$$R_{\text{se,droplet,i}} = \left[R_{\text{se,droplet,min,i}} \cdot \left(1 - \frac{\theta_{\text{se,i}}}{\theta_{\text{fu,i}}} \right) + R_{\text{se,droplet,max,i}} \cdot \left(\frac{\theta_{\text{se,i}}}{\theta_{\text{fu,i}}} - 0.5 \right) \right] \cdot 2 \tag{16.4-300}$$

For $\frac{\theta_{\text{se,i}}}{\theta_{\text{fu,i}}} \leq 0.5$

$$R_{\text{se,droplet,i}} = R_{\text{se,droplet,min,i}} \tag{16.4-301}$$

Note that the definitions 16.4-257 and 16.4-261 lead to a continuous change in $R_{\text{droplet,i}}$ across the transition from a fuel to a steel flow regime. The number of steel droplets and the heat-transfer area are obtained from formulas similar to 16.4-258 and 16.4-259. The heat-transfer coefficient is defined by:

$$H_{\text{se,fu,i}} = \frac{1}{\frac{R_{\text{se,droplet,i}}}{k_{\text{se}}} + \frac{1}{H_{\text{fu,i}}}} \tag{16.4-302}$$

where $H_{\text{fu,i}} = 3 \cdot 10^4$ and has already been explained in Eq. (16.4-281).

16.4.3.10.1.5 Heat-transfer Coefficients between Fuel and Fuel and Steel Chunks

As explained in 16.4.3.9.2, the fuel and steel chunks have a cylindrical geometry and the aspect ratio R/L can be varied via the input. Thus, it is necessary to distinguish between the heat transfer across the surface normal to the flow $A^N_{fl,\text{i}}$ and that across the surface parallel to the flow direction, $A^L_{fl,\text{i}}$. Note that the chunks are always assumed to be parallel to the flow direction, and that the total surface of the fuel chunks is given by $A^L_{fl,\text{i}} + A^N_{fl,\text{i}}$. Thus, the generalized heat-transfer coefficient between the fuel and the fuel and steel chunks can be written as:

$$H'_{\text{fu,}fl\text{,i}} = H^L_{\text{fu,}fl\text{,i}} \cdot \frac{A^L_{\text{fu,}fl\text{,i}}}{\text{AXMX} \cdot \Delta z_{\text{i}}} + H^N_{\text{fu,}fl\text{,i}} \cdot \frac{A^N_{\text{fu,}fl\text{,i}}}{\text{AXMX} \cdot \Delta z_{\text{i}}} \tag{16.4-303}$$

$$H'_{\text{fu,}sl\text{,i}} = H^L_{\text{fu,}sl\text{,i}} \cdot \frac{A^L_{\text{fu,}sl\text{,i}}}{\text{AXMX} \cdot \Delta z_{\text{i}}} + H^N_{\text{fu,}sl\text{,i}} \cdot \frac{A^N_{\text{fu,}sl\text{,i}}}{\text{AXMX} \cdot \Delta z_{\text{i}}} \tag{16.4-304}$$

For the steel flow regimes, both bubbly and annular, no contact is present between the molten fuel and chunks:

$$H'_{\text{fu,}fl\text{,i}} = H'_{\text{fu,}sl\text{,i}} = 0 \tag{16.4-305}$$

For the bubbly fuel flow regime, the heat-transfer coefficients are defined as follows:

$$H^L_{\text{fu,}fl\text{,i}} = \frac{1}{\frac{1}{H_{\text{fu,i}}} + \frac{R_{fl,\text{i}}}{k_{\text{fu}}}}; \quad H^N_{\text{fu,}fl\text{,i}} = \frac{1}{\frac{1}{H_{\text{fu,i}}} + \frac{L_{fl,\text{i}}}{2k_{\text{fu}}}} \tag{16.4-306}$$

and

$$H^L_{\text{fu,}sl\text{,i}} = \frac{1}{\frac{1}{H_{\text{fu,i}}} + \frac{R_{sl,\text{i}}}{k_{\text{se}}}}; \quad H^N_{\text{fu,}sl\text{,i}} = \frac{1}{\frac{1}{H_{\text{fu,i}}} + \frac{L_{sl,\text{i}}}{2k_{\text{se}}}} \tag{16.4-307}$$

where $H_{\text{fu,i}}$ has been defined in Eq. (16.4-281). The corresponding heat-transfer areas are:

$$A^L_{\text{fu,}fl\text{,i}} = A^L_{fl,\text{i}}; \quad A^N_{\text{fu,}fl\text{,i}} = 0.5A^N_{fl,\text{i}} \,. \tag{16.4-308}$$

$$A^L_{\text{fu,}sl\text{,i}} = A^L_{sl,\text{i}}; \quad A^N_{\text{fu,}sl\text{,i}} = 0.5A^N_{sl,\text{i}} \,. \tag{16.4-309}$$

For the annular fuel flow regime, the chunks can have velocities quite different from those of the molten fuel, and convective effects must be taken into account. The heat-transfer coefficients are still defined by Eq. (16.4-306) and Eq. (16.4-307), but $H_{\text{fu,i}}$ is replaced by $H_{\text{fu,i}} + H_{\text{fu,convective,i}}$, where $H_{\text{fu,convective,i}}$ is defined as:

$$H_{\text{fu,convective,i}} = \frac{1}{D_{\text{H,fu,i}}} \cdot \text{CIA3} \cdot \mu_{\text{fu}} \cdot C_{\text{p,fu}} \cdot \text{Re}_{\text{fu,i}}^{0.8} \tag{16.4-310}$$

where

$$\text{Re}_{lu,\text{i}} = \rho_{\text{fu}} \cdot |u_{\text{fu,i}} - u_{lu,\text{i}}| \cdot \frac{2R_{lu,\text{i}}}{\mu_{\text{fu}}} \tag{16.4-311}$$

The heat-transfer areas for the annular flow are significantly different from those used for the bubbly fuel flow. The normal surface is not in contact with the molten fuel, thus:

$$A^N_{\text{fu},fl,\text{i}} = A^L_{\text{fu},sl,\text{i}} = 0 \tag{16.4-312}$$

The lateral heat transfer areas are defined as follows:

$$\begin{aligned} A^L_{\text{fu},fl,\text{i}} = A^L_{fl,\text{i}} \cdot F_{\text{area,bd},lu,\text{i}} \cdot C_{\text{fu,cl,i}} \cdot \frac{A^L_{\text{cl,i}}}{A^L_{\text{cl,i}} + A^L_{\text{sr,i}}} \\ + C_{\text{fu,sr,i}} \cdot \frac{A^L_{\text{sr,i}}}{A^L_{\text{cl,i}} + A^L_{\text{sr,i}}} \end{aligned} \tag{16.4-313}$$

where $F_{\text{area,bd},lu,\text{i}}$ is the fraction of the area of the chunks in contact with the channel boundaries. It is obtained by assuming that chunks are uniformly distributed across the channel area and defining the contact area for the chunks near the boundary (currently the contact area is equal to $0.5 \cdot A^L_{\text{fl,i}}$). $A^L_{\text{fu},sl,\text{i}}$ is defined similarly to Eq. (16.4-313), but $A^L_{fl,\text{i}}$ is replaced by $A^L_{sl,\text{i}}$.

16.4.3.10.1.6 Heat-transfer Coefficients between the Molten Steel and Fuel and Steel Chunks

For the annular and bubbly fuel flow regimes, these coefficients are zero because the steel and the chunks are not in contact.

For the annular and bubbly steel flow regimes, the considerations made in Section 16.4.3.10.1.5 still apply and the Eq. (16.4-303) through 16.4-272 can be used, but $H_{\text{fu,i}}$ is replaced by $H_{\text{se,i}}$, defined similarly for steel, and $H_{\text{fu,convective,i}}$ is replaced by $H_{\text{se,convective,i}}$ defined below:

$$H_{\text{se,convective,i}} = \frac{1}{D_{\text{H,se,i}}} \cdot \text{CIA3} \cdot \mu_{\text{se}} \cdot C_{\text{p,se}} \cdot \text{Re}_{\text{lu,i}}^{0.8} \tag{16.4-314}$$

16.4.3.10.1.7 Heat-transfer Coefficients between Molten Steel and Cladding and Structure

The coefficients are zero for the fuel flow regimes, both annular and bubbly:

$$H'_{\text{se,cl,i}} = H'_{\text{se,sr,i}} = 0 \tag{16.4-315}$$

For the annular steel flow regime, heat-transfer coefficients are defined as follows:

$$H_{\text{se,cl,i}} = H_{\text{se,sr,i}} = \frac{1}{D_{\text{H,se,i}}} \cdot \text{CIA3} \cdot \mu_{\text{se}} \cdot C_{\text{p,se}} \cdot \text{Re}_{\text{se,i}}^{0.8} \tag{16.4-316}$$

and the respective areas, $A_{\text{se},cl,\text{i}}$ and $A_{\text{se,sr,i}}$ are identical to Eq. (16.4-285).

For the bubbly steel regime, the heat-transfer is given by:

$$H_{\text{se,cl,i}} = H_{\text{se,sr,i}} = \frac{1}{D_{\text{H,se,i}}} \cdot \text{CIA3} \cdot \mu_{\text{se}} \cdot C_{\text{p,se}} \cdot \text{Re}_{\text{se,i}}^{0.8} + \frac{4k_{\text{se}}}{D_{\text{H,se,i}}} \tag{16.4-317}$$

and the heat-transfer areas $A^L_{\text{se,cl,i}}$ and $A^L_{\text{se,sr,i}}$ are defined by Eq. (16.4-287).

16.4.3.10.1.8 Heat-transfer Coefficients between Steel and the Frozen Fuel on Cladding Structure

These coefficients are zero for the fuel annular and bubbly flow regimes:

$$H'_{\text{se,ffc,i}} = H'_{\text{se,ffs,i}} = 0 \tag{16.4-318}$$

For the steel regimes, the considerations made in Section 16.4.3.10.1.3 for the fuel regimes apply. Thus, Eq. (16.4-288) through Eq. (16.4-294) can be used to define the corresponding heat-transfer coefficients and transfer areas if one replaces $H_{\text{fu,i}}$ by $H_{\text{se,i}}$, defined as follows:

$$H_{\text{se,i}} = \frac{1}{D_{\text{H,se,i}}} \cdot \text{CIA3} \cdot \mu_{\text{se}} \cdot C_{\text{p,se}} \cdot \text{Re}_{\text{se,i}}^{0.8} \tag{16.4-319}$$

16.4.3.10.1.9 Heat Transfer between Steel and Sodium/Fission Gas

The coefficient is zero for the fuel flow regimes:

$$H'_{\text{se,Na},i} = 0 \tag{16.4-320}$$

All considerations made for the fuel flow regimes in Section 16.4.3.10.1.1 apply here and Eq. (16.4-270) through Eq. (16.4-282) should be used, if $h_{\text{fu,i}}$ is replaced by $H_{\text{se,i}}$, as appropriate. Thus, the heat-transfer coefficient for the annual steel regime becomes from Eq. (16.4-270):

$$H_{\text{se,Na,i}} = \frac{1}{\frac{1}{H_{\text{se,i}}} + \frac{1}{H_{\text{Na,i}}}} \tag{16.4-321}$$

where $H_{\text{se,i}}$ is defined by Eq. (16.4-319). The same formula applies for the bubbly steel regime from Eq. (16.4-280), but $H_{\text{se,i}}$ is defined in a manner similar to Eq. (16.4-281).

16.4.3.10.1.10 Heat-transfer Coefficients between Fuel Steel Chunks and Cladding and Structure

The chunks can be in contact with the cladding and structure only in the partial annular fuel flow regime (this is also the default when neither fuel nor steel is present in the channel). For all other regimes, the heat-transfer coefficients are equal to zero:

$$\begin{aligned} H'_{\text{cl,f}l\text{,i}} &= 0 \\ H'_{\text{cl,s}l\text{,i}} &= 0 \\ H'_{\text{sr,f}l\text{,i}} &= 0 \\ H'_{\text{sr,s}l\text{,i}} &= 0 \end{aligned}$$

For the partial annular fuel flow regime, the heat-transfer coefficients are defined below:

$$H_{\text{cl,f}l\text{,i}} = H_{\text{sr,f}l\text{,i}} = \frac{k_{\text{fu}}}{R_{l\text{u,i}}} \tag{16.4-322}$$

$$H_{\text{cl,s}l\text{,i}} = H_{\text{sr,s}l\text{,i}} = \frac{k_{\text{se}}}{R_{l\text{u,i}}} \tag{16.4-323}$$

The definition of the heat-transfer areas accounts for the fact that only the chunks close to the boundaries not covered by molten fuel exchange energy directly with the cladding and structure.

$$A_{\text{cl,f}l\text{,i}} = A^L_{\text{f}l\text{,i}} \cdot F_{\text{area,bd,}l\text{u,i}} \cdot \frac{A^L_{\text{cl,i}}}{A^L_{\text{cl,i}} + A^L_{\text{sr,i}}} \cdot (1 - C_{\text{fu,cl,i}}) \tag{16.4-324}$$

$$A_{\text{sr},fl,\text{i}} = A^L_{fl,\text{i}} \cdot F_{\text{area,bd},lu,\text{i}} \cdot \frac{A^L_{\text{sr,i}}}{A^L_{\text{cl,i}} + A^L_{\text{sr,i}}} \cdot (1 - C_{\text{fu,sr,i}}) \tag{16.4-325}$$

$$A_{\text{cl},sl,\text{i}} = A^L_{sl,\text{i}} \cdot F_{\text{area,bd},lu,\text{i}} \cdot \frac{A^L_{cl,\text{i}}}{A^L_{\text{cl,i}} + A^L_{\text{sr,i}}} \cdot (1 - C_{\text{fu,cl,i}}) \tag{16.4-326}$$

$$A_{\text{sr},sl,\text{i}} = A^L_{sl,\text{i}} \cdot F_{\text{area,bd},lu,\text{i}} \cdot \frac{A_{\text{sr,i}}}{A^L_{\text{cl,i}} + A^L_{\text{sr,i}}} \cdot (1 - C_{\text{fu,sr,i}}) \tag{16.4-327}$$

16.4.3.10.1.11 Heat-transfer between Sodium/Fission Gas Mixture and the Fuel/Steel Chunks

These coefficients are zero in the bubbly flow regimes when there is no contact between sodium and chunks.

$$H'_{\text{Na,f}l,\text{i}} = H'_{\text{Na,}sl,\text{i}} = 0 \tag{16.4-328}$$

For the annular fuel and steel regimes the generalized coefficients are defined as follows:

$$H'_{\text{Na,f}l,\text{i}} = H^L_{\text{Na,f}l,\text{i}} \cdot \frac{A^L_{\text{Na,f}l,\text{i}}}{\Delta z_\text{i} \cdot \text{AXMX}} + H^N_{\text{Na,f}l,\text{i}} \cdot \frac{A^N_{\text{Na,f}l,\text{i}}}{\Delta z_\text{i} \cdot \text{AXMX}} \tag{16.4-329}$$

$$H'_{\text{Na,}sl,\text{i}} = H^L_{\text{Na,}sl,\text{i}} \cdot \frac{A^L_{\text{Na,}sl,\text{i}}}{\Delta z_\text{i} \cdot \text{AXMX}} + H^N_{\text{Na,}sl,\text{i}} \cdot \frac{A^N_{\text{Na,}sl,\text{i}}}{\Delta z_\text{i} \cdot \text{AXMX}} \tag{16.4-330}$$

The heat-transfer coefficients in Eq. (16.4-329) and Eq. (16.4-331) are defined below:

$$H^L_{\text{Na,f}l,\text{i}} = \frac{1}{\frac{1}{H_{\text{Na,i}}} + \frac{R_{lu,\text{i}}}{k_{\text{fu}}}}; \quad H^N_{\text{Na,f}l,\text{i}} = \frac{1}{\frac{1}{H_{\text{Na,i}}} + \frac{L_{fl,\text{i}}}{2k_{\text{fu}}}}; \tag{16.4-331}$$

$$H^L_{\text{Na,}sl,\text{i}} = \frac{1}{\frac{1}{H_{\text{Na,i}}} + \frac{R_{lu,\text{i}}}{k_{\text{se}}}}; \quad H^N_{\text{Na,}sl,\text{i}} = \frac{1}{\frac{1}{H_{\text{Na,i}}} + \frac{L_{sl,\text{i}}}{2k_{\text{se}}}}; \tag{16.4-332}$$

The respective heat-transfer areas are defined as follow:

$$A^L_{\text{Na,f}l,\text{i}} = A^L_{fl,\text{i}} \cdot (1 - F_{\text{area,bd},lu,\text{i}}); \quad A^N_{\text{Na,f}l,\text{i}} = 0.5 \cdot A^N_{fl,\text{i}} \tag{16.4-333}$$

$$A^L_{\text{Na,}sl,\text{i}} = A^L_{sl,\text{i}} \cdot (1 - F_{\text{area,bd},lu,\text{i}}); \quad A^N_{\text{Na,}sl,\text{i}} = 0.5 \cdot A^N_{sl,\text{i}} \tag{16.4-334}$$

16.4.3.10.1.12 Heat-transfer between the Fuel and Steel Solid Chunks

The coefficient $H'_{fl,sl,i}$ is different from zero only when both fuel and steel chunks are present in a cell. Given the geometry of the chunks presented in Section 16.4.3.9, the heat-transfer coefficient is defined by:

$$H_{fl,sl,\text{i}} = \frac{1}{\frac{L_{fl,\text{i}}}{2k_{\text{fu}}} + \frac{L_{sl,\text{i}}}{2k_{\text{se}}}} \tag{16.4-335}$$

and the heat-transfer area by:

$$A_{fl,sl,\text{i}} = 0.5 \cdot A^N_{fl,\text{i}} \tag{16.4-336}$$

16.4.3.10.1.13 Heat Transfer between Fuel/Steel Chunks and the Frozen Fuel on Cladding and Structure

The coefficients $H'_{fl,\text{ffc,i}}$, $H'_{fl,\text{ffs,i}}$, $H'_{sl,\text{ffc,i}}$, and $H'_{sl,\text{ffs,i}}$ are non-zero only if there is little molten fuel and/or steel present, i.e.:

$$\frac{\theta_{\text{fu,i}} + \theta_{\text{se,i}}}{\theta_{\text{ch,op,i}} - \theta_{l\text{u,i}}} \leq 0.1 \tag{16.4-337}$$

and the frozen fuel crust covers a significant fraction of the channel perimeter, i.e.:

$$C_{\text{ff,cl,i}} \geq 0.1 \text{ or } C_{\text{ff,sr,i}} \geq 0.1 \tag{16.4-338}$$

If the above conditions are satisfied, the heat-transfer coefficients are defined as follows:

$$H_{fl,\text{ffc,i}} = \frac{1}{\frac{l_{\text{ff,cl,i}}}{2k_{\text{fu}}} + \frac{R_{l\text{u,i}}}{k_{\text{fu}}}}; \quad H_{sl,\text{ffc,i}} = \frac{1}{\frac{l_{\text{ff,cl,i}}}{2k_{\text{fu}}} + \frac{R_{l\text{u,i}}}{k_{\text{se}}}} \tag{16.4-339}$$

$$H_{fl,\text{ffs,i}} = \frac{1}{\frac{l_{\text{ff,sr,i}}}{2k_{\text{fu}}} + \frac{R_{l\text{u,i}}}{k_{\text{fu}}}}; \quad H_{sl,\text{ffs,i}} = \frac{1}{\frac{l_{\text{ff,sr,i}}}{2k_{\text{fu}}} + \frac{R_{l\text{u,i}}}{k_{\text{se}}}} \tag{16.4-340}$$

The respective heat-transfer areas are defined below:

$$A_{fl,\text{ffc,i}} = A^L_{fl,\text{i}} \cdot F_{\text{area,bd},l\text{u,i}} \cdot \frac{A^L_{\text{cl,i}}}{A^L_{\text{cl,i}} + A^L_{\text{sr,i}}} \cdot C_{\text{ff,cl,i}} \tag{16.4-341}$$

$$A_{fl,\text{ffs,i}} = A^L_{fl,\text{i}} \cdot F_{\text{area,bd},l\text{u,i}} \cdot \frac{A^L_{\text{sr,i}}}{A^L_{\text{cl,i}} + A^L_{\text{sr,i}}} \cdot C_{\text{ff,sr,i}} \tag{16.4-342}$$

The areas for the steel chunks, $A_{sl,\text{ffc,i}}$ and $A_{sl,\text{ffs,i}}$ can be obtained from the above definitions by replacing $A^L_{fl,\text{i}}$ with $A^L_{sl,\text{i}}$.

16.4.3.10.1.14 Heat-transfer Coefficients between Sodium/Fission Gas and Cladding or Structure

The coefficients $H'_{\text{Na,cl,i}}$ and $H'_{\text{Na,sr,i}}$ are non-zero only for the partial annular fuel flow regime. In all other regimes, the sodium/fission gas mixture is not in direct contact with the cladding and structure. The heat-transfer coefficient is calculated as follows:

If $\alpha_{\text{Na,i}} \leq 0.15$,

$$H_{\text{Na,cl,i}} = H_{\text{Na,sr,i}} = \left[\text{C1}\left(D_{\text{H,Mi,i}} \cdot \rho_{\text{N1}} \cdot |u_{\text{Mi,i}} + u_{\text{Mi,i+1}} \cdot C_{\text{Mo,Mi,i+1}}| \cdot 0.5 \cdot \frac{C_{\text{p,N1}}}{k_{\text{N1}}}\right)^{\text{C2}} + \text{C3}\right] \cdot \frac{k_{\text{N1}}}{D_{\text{H,Mi,i}}} \tag{16.4-343}$$

The value of k_{N1} is given by the input constant CDNL. The constants $C1$, $C2$ and $C3$ are the source input constants used in the preboiling model.

If $\alpha_{\text{Na,i}} > 0.15$ and

a) $T_{\text{Na,i}} \leq T_{\text{cl,i}}$ (boiling situation)

$$H_{\text{Na,cl,i}} = \begin{cases} H_{\text{Na,boiling}} = \text{CFNAEV} & \text{for } 1.5 < \alpha < 0.5 \\ [H_{\text{Na,boiling}} \cdot (1 - \alpha_{\text{Na,i}}) + H_{\text{Nv,i}} & \\ \cdot (\alpha_{\text{Na,i}} - 0.5)] \cdot 2 & \text{for } 0.5 \leq \alpha < 1 \\ H_{\text{NV,i}} = \left[\text{C1} \cdot \left(D_{\text{H,Mi,i}} \cdot \rho_{\text{Mi,i}} \cdot |u_{\text{Mi,i}}| \cdot \frac{C_{\text{p,Mi}}}{k_{\text{Mi}}}\right)^{\text{C2}} + \text{C3}\right] \cdot \frac{k_{\text{Mi}}}{D_{\text{H,Mi,i}}} & \text{for } \alpha = 1 \end{cases} \tag{16.4-344}$$

The value of k_{Mi} is given by the input constant CDVG.

b) $T_{\text{Na,i}} > T_{\text{cl,i}}$ (condensation situation)

$$H_{\text{Na,cl,i}} = \begin{cases} H_{\text{Nv,i}} \text{ as defined in Eq.16.4-303c} & \text{for } \alpha_{\text{Na,i}} = 1 \\ & \text{and } T_{\text{cl,i}} < T_{\text{sat,Na}}\,(P_{\text{Na,i}}) \\ \text{CFNACN} \cdot \frac{\rho'_{\text{Na}}}{\rho'_{\text{Na}} + \rho'_{\text{fi}}} + H_{\text{Nv}} & \text{for } \alpha_{\text{Na,i}} = 1 \\ \cdot \frac{\rho'_{\text{fi}}}{\rho'_{\text{Na}} + \rho'_{\text{fi}}} & \text{and } T_{\text{cl,i}} \geq T_{\text{sat,Na}}\,(P_{\text{Na,i}}) \\ H_{\text{Na,cond}} = \text{CFNACN} & \text{for } 0.5 \leq \alpha_{\text{Na,i}} < 1 \\ \left[H_{\text{Na,cond}} \cdot \alpha_{\text{Na,i}} + H_{\text{Na,cond}} \cdot 10^{-2}\right. & \\ \left.\cdot (0.5 - \alpha_{\text{Na,i}})\right] \cdot 2 & \text{for } 0.15 < \alpha < 0.5 \end{cases} \tag{16.4-345}$$

It is noted that the branch Eq. (16.4-345) has not been implemented in the initial release version but exists in the chunk version.

For $\alpha_{\text{Na,i}} > .15$, the heat-transfer coefficient between sodium and structure is obtained from Eq. (16.4-344) for $T_{\text{Na,i}} \leq T_{\text{sr,i}}$ and from Eq. (16.4-345) for $T_{\text{Na,i}} > T_{\text{sr,i}}$.

The respective heat-transfer areas are defined as follows:

$$A^L_{\text{Na,cl,i}} = A^L_{\text{cl,i}} \cdot (1 - C_{\text{fu,cl,i}}) \cdot (1 - C_{\text{lu,cl,i}}) \tag{16.4-346}$$

and

$$A^L_{\text{Na,sr,i}} = A^L_{\text{sr,i}} \cdot (1 - C_{\text{fu,sr,i}}) \cdot (1 - C_{\text{lu,sr,i}}) \tag{16.4-347}$$

16.4.3.10.1.15 Heat-transfer Coefficients between Sodium and Frozen Fuel on Cladding and Structure

The coefficients $H'_{\text{ffc,Na,i}}$ and $H'_{\text{ffs,Na,i}}$ are non-zero only in the partial annular flow regime. For this regime, the heat transfer coefficients are defined below:

$$H_{\text{ffc,Na,i}} = \frac{1}{\frac{1}{H_{\text{Na,i}}} + \frac{0.5 \cdot l_{\text{ff,cl,i}}}{k_{\text{fu}}}} \tag{16.4-348}$$

$$H_{\text{ffs,Na,i}} = \frac{1}{\frac{1}{H_{\text{Na,i}}} + \frac{0.5 \cdot l_{\text{ff,sr,i}}}{k_{\text{fu}}}} \tag{16.4-349}$$

where $H_{\text{Na,i}}$ is defined by Eq. (16.4-272).

The respective heat-transfer areas are defined as follows:

$$A^L_{\text{ffc,Na,i}} = A^L_{\text{cl,i}} \cdot C_{\text{ff,cl,i}} \cdot (1 - C_{\text{fu,ff,cl,i}}) \tag{16.4-350}$$

$$A^L_{\text{ffs,Na,i}} = A^L_{\text{sr,i}} \cdot C_{\text{ff,sr,i}} \cdot (1 - C_{\text{fu,ff,sr,i}}) \tag{16.4-351}$$

16.4.3.10.2 Friction Coefficients Used in the Momentum Equations

In Section 16.4.3.8, which describes the momentum equations, several friction coefficients used in the momentum source terms were left undefined. These coefficients are calculated in the routine LETRAN and are presented in this section.

16.4.3.10.2.1 Friction Factor Between Mixture and Cladding/Structure

The friction factor $f_{\text{Mi,i}}$ is referenced in Eq. (16.4-166). It is zero for all flow regimes except the partial annual fuel flow regime. For this flow regime, it is defined as follows:

$$f_{\text{Mi,i}} = \text{AFRV} \cdot \text{Re}_{\text{Mi,i}}^{\text{BFRV}} \cdot C_{\text{Mi,bd,i}} \tag{16.4-352}$$

where AFRV, BFRV are input constants and $C_{\text{Mi,bd,i}}$ is the fraction of the channel boundary (cladding + structure + crust) which is in direct contact with the gas mixture in cell i. It is defined as:

$$\begin{aligned} C_{\text{Mi,bd,i}} = \big[& A^L_{\text{cl,i}} \cdot (1 - C_{\text{mfu,cl,i}} - C f_{\text{ff,cl,i}} \cdot C_{\text{fu,ff,cl,i}}) \\ & + A^L_{\text{sr,i}} \cdot (1 - C_{\text{mfu,sr,i}} - C_{\text{ff,sr,i}} \cdot C_{\text{fu,ff,sr,i}}) \big] \\ & \Big/ \left(A^L_{\text{cl,i}} + A^L_{\text{sr,i}} \right) \end{aligned} \tag{16.4-353}$$

where

$$C_{\text{mfu,cl,i}} = C_{\text{fu,cl,i}} - C_{\text{ff,cl,i}} \tag{16.4-354}$$

and

$$C_{\text{mfu,sr,i}} = C_{\text{fu,sr,i}} - C_{\text{ff,sr,i}} \tag{16.4-355}$$

16.4.3.10.2.2 Friction Factors Between Mixture and Fuel

The friction factors $f_{\text{Mi,fu,i}}$ and $f_{\text{Mi,se,i}}$ are referenced in Eq. (16.4-168). The factor $f_{\text{Mi,fu,i}}$ is used only in the partial annular fuel flow regime and is defined below:

$$f_{\text{Mi,fu,i}} = \text{AFRV} \cdot \text{Re}_{\text{Mi,fu,i}}^{\text{BFRV}} \cdot C_{\text{Mi,fu,i}} \tag{16.4-356}$$

$\text{Re}_{\text{Mi,fu,i}}$ is defined as:

$$\text{Re}_{\text{Mi,fu,i}} = D_{\text{H,Mi,i}} \cdot |u_{\text{MI,i}} - u_{\text{fu,i}}| \cdot \rho_{\text{Mi,i}} \cdot \frac{1}{\mu_{\text{Mi,i}}} \tag{16.4-357}$$

and $C_{\text{Mi,fu,i}}$, the fraction of the mixture perimeter in contact with the molten fuel is given by:

$$\begin{aligned} C_{\text{Mi,fu,i}} = & \left[A^L_{\text{cl}} \cdot (C_{\text{mfu,cl,i}} + C_{\text{ff,cl,i}} \cdot C_{\text{fu,ff,cl,i}}) \right. \\ & \left. + A^L_{\text{sr}} \cdot (C_{\text{mfu,sr,i}} + C_{\text{ff,sr,i}} \cdot C_{\text{fu,ff,sr,i}}) \right] \Big/ \left(A^L_{\text{cl,i}} + A^L_{\text{sr,i}} \right) \end{aligned} \tag{16.4-358}$$

The factor $f_{\text{Mi,se,i}}$ is used in the annular steel flow regime and takes into account the possibility of flooding. It is defined as follows:

$$f_{\text{Mi,se,i}} = \text{AFRV} \cdot \text{Re}_{\text{Mi,fu,i}}^{\text{BRFV}} \cdot C_{\text{FLOOD,i}} \tag{16.4-359}$$

Where the flooding coefficient depends on the velocity ratio V_{RATIO}:

$$V_{\text{RATIO,i}} = \frac{(U_{\text{Mi,i}} + u_{\text{Mi,i+1}} \cdot C_{\text{Mo,Mi,i+1}}) \cdot 0.5}{u_{\text{FLOOD,i}}} \tag{16.4-360}$$

$$u_{\text{FLOOD,i}} - 2.3\sqrt{\frac{\rho_{\text{se,i}}}{\rho_{\text{Mi,i}}} \cdot \frac{2g \cdot l_{\text{se,ch,i}}}{\text{AFRV} \cdot \text{Re}_{\text{Mi,fu,i}}^{\text{BFRV}} \cdot F_{\text{FLOOD}}\left(\alpha_{\text{Na,i}}\right)}} \tag{16.4-361}$$

$$F_{\text{FLOOD}}\left(\alpha_{\text{Na,i}}\right) = \begin{cases} 1 + 75 \cdot \left(1 - \alpha_{\text{Na,i}}\right) & 0.5 < \alpha \leq 1 \\ 38.5 & \alpha \leq 0.5 \end{cases} \tag{16.4-362}$$

The value of $C_{\text{FLOOD,i}}$ is also dependent on the previous flooding history as shown below:

If $C_{\text{FLODD,i}}^{n} \leq 1$:

$$C_{\text{FLODD,i}}^{n} = \begin{cases} F_{\text{FLOOD}}\left(\alpha_{\text{Na,i}}\right) & \text{if } V_{\text{RATIO,i}} \geq 1.1 \\ C_{\text{FLODD,i}}^{n} & \text{otherwise} \end{cases} \tag{16.4-363}$$

If $C_{\text{FLODD,i}}^{n} > 1$

$$C_{\text{FLODD,i}}^{n+1} = \begin{cases} 1 & \text{if } V_{\text{RATIO}} < 0.9 \\ C_{\text{FLODD,i}}^{n} & \text{otherwise} \end{cases} \tag{16.4-364}$$

16.4.3.10.2.3 Area Coefficients Used in the Momentum Source Terms

The area coefficient $C_{\text{AREA,Mi,fu,i}}$ has been introduced in Eq. (16.4-173). It defines the fraction of the lateral area of chunks in contract with the mixture and is zero for the fuel and steel bubbly flow regimes. For the annular flow regimes, it is calculated as follows:

$$C_{\text{AREA,Mi,fu,i}} = \left(1 - F_{\text{area,bd,fu,i}}\right) \tag{16.4-365}$$

The area coefficients $C_{\text{AREA,se,i}}$ and $C_{\text{AREA,fu,i}}$ have been introduced in Eq. (16.4-196) and Eq. (16.4-197). They account for the fact that only a fraction of the fuel steel perimeter is in contact with the stationary walls. They are defined below:

For the fuel bubbly flow regime:

$$C_{\text{AREA,fu,i}} = 1; \quad C_{\text{AREA,se,i}} = 0 \tag{16.4-366}$$

For the steel bubbly flow regime:

$$C_{\text{AREA,fu,i}} = 0; \quad C_{\text{AREA,se,i}} = 1 \tag{16.4-367}$$

For the annular steel flow regime:

$$C_{\text{AREA,fu,i}} = 0; \quad C_{\text{AREA,se,i}} = 1 \tag{16.4-368}$$

For the partial annular fuel flow regime:

$$\begin{aligned} C_{\text{AREA,fu,i}} = &\left[A_{\text{cl,i}}^{L} \cdot \left(C_{\text{mfu,cl,i}} + C_{\text{ff,cl,i}} \cdot C_{\text{fu,ff,cl,i}}\right)\right. \\ &\left. + A_{\text{sr,i}}^{L} \cdot \left(C_{\text{mfu,sr,i}} + C_{\text{ff,sr,i}} \cdot C_{\text{fu,ff,sr,i}}\right)\right] / \left(A_{\text{cl,i}}^{L} + A_{\text{sr,i}}^{L}\right) \\ C_{\text{AREA,se,i}} = &\ 0 \end{aligned} \tag{16.4-369}$$

16.4.3.11 Time Step Determination for the Channel Hydrodynamic Model

This section describes the calculation of the maximum time step for the coolant channel hydrodynamic model. This time step will be used in the determination of the LEVITATE time step as outlined in Section 16.7.3. The maximum channel time step is calculated using the sonic Courant condition for the multicomponent channel flow. Only a fraction of the Courant calculated time is used, to account for inaccuracies in the sonic velocity calculation. Thus:

$$\Delta t_{\text{LE,ch}} = 0.4 \cdot \min \left[\frac{\Delta z_{\text{i}}}{(u_{\text{sonic,i}} + |u_{\text{Mi,i}}|)} \right]_{\text{i=IFMIBT,IFMITP}} \tag{16.4-370}$$

The minimum in Eq. (16.4-370) is evaluated over all axial cells of the interaction region. The sonic velocity in each cell is calculated using an expression [16-14] for an adiabatic homogeneous two-phase mixture of liquid sodium and fission gas/sodium vapor. The compressibility of liquid fuel is much smaller than that of liquid sodium and thus the fuel is assumed to be incompressible in the calculation of the sonic velocity in the channel. The effect of fuel and steel vapor is not included.

$$\begin{aligned} u^2_{\text{sonic,i}} = \gamma_{\text{vg}} \cdot (P_{\text{fi,i}} + P_{\text{Na,i}}) \,/\, \{ \alpha^2_{\text{vg,i}} (\rho_{\text{fi}} + \rho_{\text{Nv}}) + \alpha_{\text{vg,i}} \cdot (1 - \alpha_{\text{vg,i}}) \\ \cdot \rho_{\text{N1}} + \left[\alpha_{\text{vg,i}} \cdot (1 - \alpha_{\text{vg,i}}) \cdot (\rho_{\text{fi}} + \rho_{\text{Nv}}) + (1 - \alpha_{\text{vg,i}})^2 \right. \\ \left. \cdot \rho_{\text{N1}} \right] \cdot \gamma_{\text{vg}} \cdot (P_{\text{fi}} + P_{\text{Na}}) \cdot \text{CMNL} \} \end{aligned} \tag{16.4-371}$$

where:

$$\alpha_{\text{vg,i}} = \frac{\theta_{\text{vg,i}}}{\theta_{\text{ch,op,i}} - \theta_{\text{fu,i}} - \theta_{\text{se,i}} - \theta_{\text{lu,i}}} \tag{16.4-372}$$

γ_{vg} = ratio of specific heat at constant pressure to that at constant volume for the fission gas/sodium vapor mixture. A value of 1.4 is used in the LEVITATE code

CMNL = compressibility of liquid sodium, an input constant

The fission-gas pressure P_{fi} and the sodium pressure P_{Na} are obtained as explained in Section 16.4.3.5.

16.5 Freezing, Melting and Heat-Transfer Processes

16.5.1 Physical Models

The models presented in this section describe a series of physical processes related to the stationary pin stubs and the hexcan wall. Except for the heat transfer in the cladding and in the hexcan wall, the processes described lead to geometry changes that affect directly the hydrodynamic model of the coolant channel.

One of the most important phenomena during a LOF accident is the freezing of the initially molten fuel and the formation of the fuel crusts. The molten fuel ejected from the pin cavity is accelerated in the coolant channel by the local pressure gradients and begins to move toward the extremities of the channel. In the process, however, it exchanges heat with the sodium, cladding and structure, all of which have a temperature well below the fuel freezing temperature. Eventually the fuel will begin to freeze, forming stationary fuel crusts on the cladding and/or structure or leading to the formation of solid fuel chunks which continue to move in the channel. Steel freezing can also occur occasionally, particularly at locations where the molten steel is the dominant component and is in contact with cold cladding or structure. The freezing of steel leads, in LEVITATE, to a local increase in the thickness of the cladding and/or structure, rather than to the formation of a distinct steel crust.

The fuel crusts formed at various axial locations can begin to melt if the power level increases, thus releasing the fuel again and allowing it to move in the coolant channel. Another mechanism which could also release the

fuel crust is the crust breakup, which can occur when the underlying steel support melts and the crust becomes unstable. The continuous heating of the cladding and hexcan wall leads eventually to steel melting. The molten steel is ablated, becoming part of the moving components in the channel, and the thickness of the cladding and/or structure is reduced, increasing the flow area of the coolant channel.

In a similar manner, the solid fuel and steel chunks generated via the fuel-pin disruption can begin to melt due to the direct heating and heat exchange with other components. This will result in a transfer of fuel from the chunk field to the molten fuel field in the coolant channel. This transfer of mass is accompanied by a transfer of energy and momentum as well as a change in the geometry of the solid chunks.

16.5.2 Description of the Method of Solution and Logic Flow

The heat-transfer equations in the cladding and structure, which are not directly related to a geometry change, are solved in the LESOEN routine, i.e., while advancing the solution for the hydrodynamic model in the channel. All other processes related to the stationary pin stubs and to the hexcan wall involve a change in the local geometry and the routines describing them are called after completing the solution for both the in-pin and coolant channel hydrodynamic models. All melting/freezing processes modeled here affect the hydrodynamic parameters of the coolant channel. Changes in the local temperatures, velocities, or pressures are performed in each of the routines described below, whenever necessary. The routine LELUME (**LE**VITATE CHUNK-**LU** **ME**LTING) calculates the melting and the size changes of the solid fuel/steel chunks at all axial locations. The routine LEFREZ (**LE**VITATE **FR**EE**Z**ING AND MELTING) then models a series of important processes, such as fuel/steel freezing and crust formation, fuel/steel chunk formation, fuel crust remelting and breakup. Next called is the routine LEABLA (**LE**VITATE ABLATION) which calculates the gradual melting and ablation of the cladding and hexcan wall. The routine LEDISR (**LE**VITATE **DISR**UPTION) performs the disruption of the fuel pin, whenever a disrupt node is predicted. The disruption, which can occur in one or more nodes in any time step, leads to changes in geometry, mass, energy and pressure for various components present in the respective cell. Finally, the routine LESRME (**LE**VITATE **S**T**R**UCTURE **ME**LTING) calculates the rupture of the hexcan wall due to melting and/or pressure burst effects.

16.5.3 Fuel/steel Freezing and Crust Formation

The calculation of fuel/steel freezing is performed in the routine LEFREZ. The fuel flow regimes lead to a situation very different from the steel flow regimes in terms of freezing and are treated separately.

16.5.3.1 The Freezing Process when a Fuel Flow Regime is Present

In this case, the molten steel component exists in the form of droplets imbedded in the molten fuel and has a temperature well above freezing. Thus, only the freezing of the molten fuel must be considered. The freezing calculation decides what amount of fuel, if any, will freeze in each node during the current time step. It also has to decide where the frozen fuel will go, when leaving the moving molten-fuel field. Three possibilities exist:

1. The freezing fuel will form (or add to) a crust on the cladding; (when the cladding has been completely ablated, the fuel can freeze on the bare fuel pins).
2. The freezing fuel will form (or add to) a crust on the hexcan wall (Figure 16.5.1).
3. The freezing fuel will generate solid fuel chunks, which will be added to the chunk field

The freezing calculation begins by examining the enthalpy of the molten fuel:

1. If $h_{\text{fu,i}} > h_{\text{fu,freeze}}$, no fuel freezing occurs. The enthalpy $h_{\text{fu,freeze}}$ is an input parameter, i.e., $h_{\text{fu,freeze}} =$ EGBBLY. It has to satisfy the conditions:

$$h_{\text{fu,so}} < h_{\text{fu,freeze}} < h_{\text{fu,liq}} \tag{16.5-1}$$

1. If $h_{\text{fu,i}} < h_{\text{fu,so}}$, rapid fuel freezing occurs, leading to the formation of solid fuel chunks only. In order to avoid numerical problems, only one tenth of the fuel mass $(0.1 \cdot \rho_{\text{fu,i}} \cdot A_{\text{fu,i}} \cdot \Delta z_{\text{i}})$ in the cell is allowed to freeze in each time step under these circumstances.
2. If $h_{\text{fu,so}} < h_{\text{fu,i}} < h_{\text{fu,freeze}}$, only partial fuel freezing is allowed to occur. The amount of freezing fuel is determined as follows:

$$\Delta\rho'_{\text{fu,i}} = \rho'_{\text{fu,i}} \cdot \frac{h_{\text{fu,freeze}} - h_{\text{fu,i}}}{h_{\text{fu,freeze}} - h_{\text{fu,so}}} \tag{16.5-2}$$

The area occupied by this newly frozen fuel is:

$$\Delta A_{\text{fu,i}} = \frac{\Delta\rho'_{\text{fu,i}}}{\rho_{\text{fu,so}}} \cdot \text{AXMX} \tag{16.5-3}$$

The assumption is made that the fuel looses most of he energy at the channel boundaries, i.e., by exchanging heat with the cladding and hexcan wall. Thus, the frozen fuel is first used to form a fuel crust on the solid boundaries. The distribution of $\Delta\rho'_{\text{fu,i}}$ between the clad and hexcan wall is made as follow:

$$\Delta\rho'_{\text{fu,cl,i}} = \Delta\rho'_{\text{fu,i}} \cdot \frac{\text{WTCL}}{\text{WTCL} + \text{WTSR}} \tag{16.5-4}$$

$$\Delta\rho'_{\text{fu,sr,i}} = \Delta\rho'_{\text{fu,i}} \cdot \frac{\text{WTSR}}{\text{WTCL} + \text{WTSR}} \tag{16.5-5}$$

where the weights WTCL and WTSR are:

$$\text{WTCL} = A^{L}_{\text{cl,i}} \cdot [(T_{\text{fu,i}} - T_{\text{cl,i}}) \cdot C_{\text{mfu,cl,i}} + (T_{\text{fu,i}} - T_{\text{ffc,i}}) \cdot C_{\text{ff,cl,i}}] \tag{16.5-6}$$

$$\text{WTSR} = A^{L}_{\text{sr,i}} \cdot [(T_{\text{fu,i}} - T_{\text{sr,i}}) \cdot C_{\text{mfu,sr,i}} + (T_{\text{fu,i}} - T_{\text{ffs,i}}) \cdot C_{\text{ff,sr,i}}] \tag{16.5-7}$$

Occasionally, one or both of these weights can be set to zero. This happens for WTCL when the cladding surface is molten and thus no solid support for freezing exists or when the pins are totally disrupted in the cell considered. In these cases, WTCL $= 0$.

Similarly, WTSR $= 0$ whenever the hexcan wall surface is molten and no crust formation can occur. In this case, we will still use Eq. (16.5-4) and 16.5-5 to calculate the distribution of the frozen fuel between cladding and hexcan wall. The situation can arise, however, where the amount of frozen fuel is too large for the available freezing area. For example, let us assume that the cladding surface is molten and WTCL $= 0$. In this case, we can have:

$$\Delta A_{\text{fu,i}} > A_{\text{ch,op,sr}} \tag{16.5-8}$$

where

$$A_{\text{ch,op,sr}} = A_{\text{ch,sr}} - A_{\text{ff,sr}} \tag{16.5-9}$$

$$A_{\text{ff,sr}} = L_{\text{ff,sr}} \cdot l_{\text{ff,sr}} \tag{16.5-10}$$

and the elements used in Eq. (16.5-7) and Eq. (16.5-8) have been defined in Section 16.4.3.9. If Eq. (16.5-6) is satisfied, only an amount of fuel consistent with the available area is allowed to freeze on the structure, while the remaining fuel generates solid chunks:

$$\Delta\rho'_{\text{fu,sr,i}} = \frac{\rho_{\text{fu,so}} \cdot A_{\text{ch,op,sr}}}{\text{AXMX}} \tag{16.5-11}$$

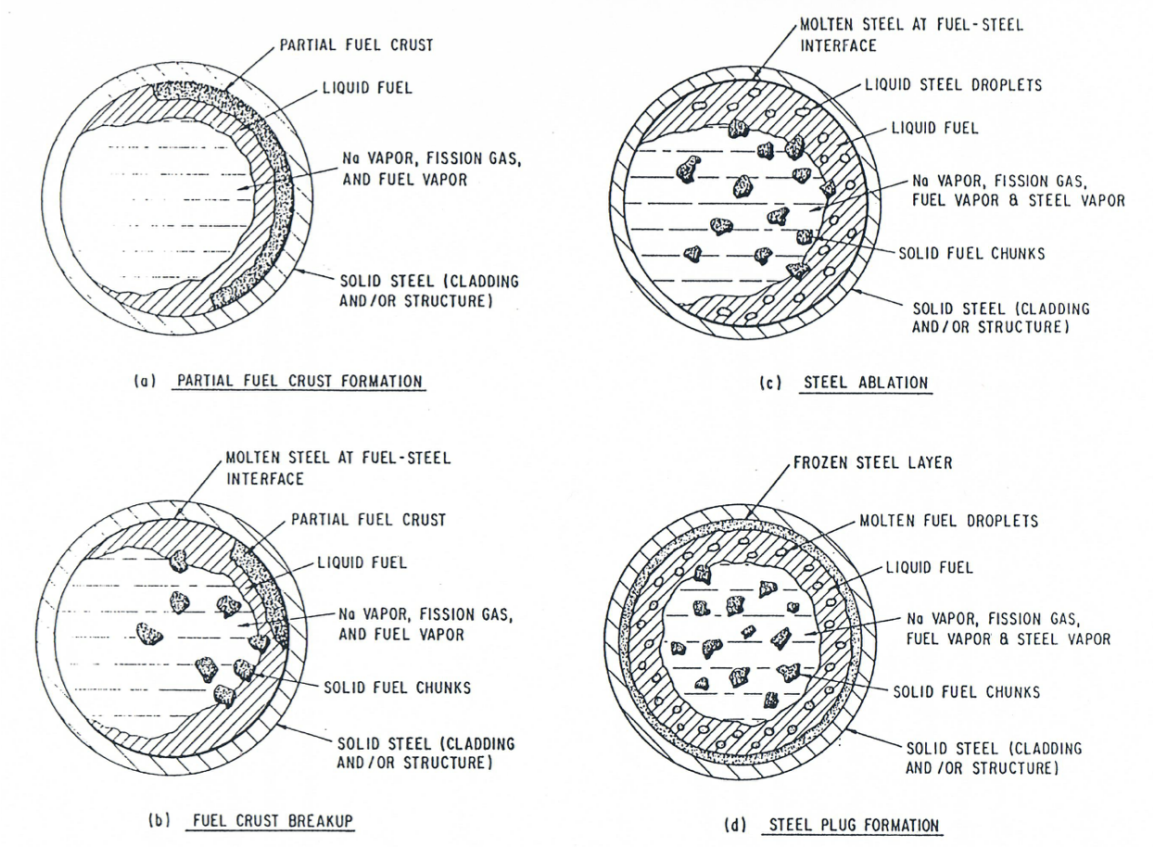


Figure 16.5.1: Freezing, Melting and Crust Breakup Processes

$$\Delta\rho'_{\text{fu,lu,i}} = \Delta\rho'_{\text{fu,i}} - \Delta\rho'_{\text{fu,sr,i}} \tag{16.5-12}$$

When both WTCL $= 0$ and WSTR $= 0$ all the freezing fuel is used to generate solid fuel chunks:

$$\Delta\rho'_{\text{fu,}l\text{u,i}} = \Delta\rho'_{\text{fu,i}} \tag{16.5-13}$$

Another decision that has to be made in connection with the fuel freezing is the steel entrapment in the frozen fuel. Based on experimental evidence obtained in the posttest examination of several fuel motion tests, it appears that only small steel droplets are trapped in the frozen fuel. LEVITATE assumes that small droplets exist only when small amounts of steel are locally present. Otherwise, the steel exists in large droplets that will tend to separate from the freezing fuel. Thus, if:

$$\frac{\theta_{\text{se,i}}}{\theta_{\text{fu,i}}} > 0.1 \tag{16.5-14}$$

no steel entrapment occurs. Otherwise, the amount of steel trapped in the frozen fuel is calculated from:

$$\Delta\rho'_{\text{se,i}} = \rho'_{\text{se,i}} \cdot \frac{\Delta\rho'_{\text{fu,i}}}{\rho'_{\text{fu,i}}} \tag{16.5-15}$$

The actual geometry of the fuel crust (i.e. thickness and area coefficient) is not changed in LEFREZ when fuel freezing occurs. The geometry is changed in LEGEOM at the beginning of the next time step. The geometry of the chunks is changed in LEFREZ due to the condition of the new chunks. The radius of the resulting chunks is a mass-weighted average of the chunks being combined. A more detailed discussion about the chunk geometry can be found in Section 16.4.3.9.

16.5.3.2 The Freezing Process when a Steel Flow Regime is Present

In a steel flow regime, both steel freezing and fuel freezing can occur. The molten fuel is in the form of droplets imbedded in the molten steel and can freeze as it approaches the steel temperature. The amount of fuel freezing in cell i, $\Delta\rho'_{\text{fu,i}}$ is calculated using the same procedure outline in Section 16.5.3.1. However, because no direct contact between the freezing fuel and the channel boundaries exists, crust formation cannot occur. All freezing fuel is used to generate solid fuel chunks:

$$\Delta\rho'_{\text{fu,}l\text{u,i}} = \Delta\rho'_{\text{fu,i}} \tag{16.5-16}$$

The steel-freezing calculation follows the same lines as the fuel calculation, but is keyed to the steel temperature rather than enthalpy:

1. If $T_{\text{se,i}} > T_{\text{se,freeze}}$, no steel freezing occurs. The temperature $T_{\text{se,freeze}}$ is determined using the input parameter FRMRSE:

$$T_{\text{se,freeze}} = T_{\text{se,so}} + (T_{\text{se,}l\text{q}} - T_{\text{se,so}}) \cdot \text{FRMRSE} \tag{16.5-17}$$

1. If $T_{\text{se,i}} < T_{\text{se,so}}$, rapid steel freezing occurs, leading to the formation of steel chunks only.
2. If $T_{\text{se,so}} < T_{\text{se,i}} < T_{\text{se,freeze}}$, only partial steel freezing is allowed to occur. The amount of freezing steel is calculated from:

$$\Delta\rho'_{\text{se,i}} = \rho'_{\text{se,i}} \cdot \frac{T_{\text{se,freeze}} - T_{\text{se,i}}}{T_{\text{se,freeze}} - T_{\text{se,so}}} \tag{16.5-18}$$

The steel can freeze on the cladding and/or on the hexcan wall, leading to an increase in the thickness of these structures. This increase is uniformly distributed along the perimeter, as opposed to the partial crust formed initially in the fuel flow regimes (Figure 16.5.1). The distribution of $\Delta\rho'_{\text{sei}}$, follows the same procedure presented in Section 16.5.3 for fuel. If not all the freezing steel can be distributed between cladding and structure (e.g., in a cell where the pins have been disrupted and/or the hexcan wall surface is molten), part or all of the freezing steel is used to generate solid steel chunks, as appropriate.

16.5.4 Fuel Crust Breakup and Remelting

The fuel crust formation on the cladding and the fuel crust formed on the hexcan wall are completely independent. Either one can be present in any axial cell, having its own temperature and geometry. The formation of the crusts was explained in Section 16.5.3. Once present, the fuel crust can disappear in two ways: via breakup and via remelting. These processes can occur in any flow regime modeled by LEVITATE.

16.5.4.1 Fuel Crust Breakup

It is assumed that the frozen fuel crust is unstable and is allowed to break up whenever there is no underlying solid support. In general, this means that the underlying steel is molten. However, when fuel crusts are present on bare fuel pins, the underlying support is also fuel. Thus, the breakup of the crust on the clad will occur when:

$$T_{\text{cl,os}} > T_{\text{se,}lq} \tag{16.5-19}$$

In this case, a gradual but fairly rapid breakup occurs. The crust is reduced each time step by:

$$\Delta\rho'_{\text{ffc}} = \rho'_{\text{ffc}} \cdot \text{CIBREK} \tag{16.5-20}$$

where CIBREK $= 0.01$ is a built-in constant, defined in the routine LEFREZ. Both the thickness $1_{\text{ff,cl,i}}$ and the length (i.e., the area coefficient $C_{\text{ff,}cl\text{,i}}$) are reduced proportionately:

$$1_{\text{ff,cl,i}} = 1_{\text{ff,cl,i}} \cdot \sqrt{1 - \text{CIBREK}} \tag{16.5-21}$$

$$C_{\text{ff,cl,i}} = C_{\text{ff,cl,i}} \cdot \sqrt{1 - \text{CIBREK}} \tag{16.5-22}$$

The fuel breaking loose from the crust generates chunks, which are then merged if necessary with the chunks already present in the cell. If steel inclusions are present in the crust, they are also reduced proportionately and added to the steel chunk field. A similar procedure is used for the structure crust breakup. The process is illustrated in Figure 16.5.1.

16.5.4.2 Remelting of the Fuel Crust

If the fuel crusts are present in a cell and breakup does not occur, the crust can still disappear via remelting. Melting of the cladding crust occurs if:

$$T_{\text{ffc,i}} > T_{\text{fu,melt}} \tag{16.5-23}$$

where

$$T_{\text{fu,melt}} = (T_{\text{fu,so}} + T_{\text{fu,}lq}) \cdot 0.5 \tag{16.5-24}$$

The amount of fuel crust melting in one time step is given by:

$$\Delta\rho'_{\text{ffc,i}} = \rho'_{\text{ffc,i}} \cdot \frac{T_{\text{ffc,i}} - T_{\text{fu,melt}}}{T_{\text{fu,}lq} - T_{\text{fu,melt}}} \tag{16.5-25}$$

If steel inclusions are present, they are reduced in the same proportion. The resulting molten fuel and steel are added to the molten fuel and steel fields in the channel, respectively.

A similar procedure is used for the remelting of the fuel crust on the hexcan wall.

16.5.5 Cladding and Hexcan Ablation

When the coolant channel contains molten fuel, which is in an annular or bubbly flow regime, the temperature at the cladding surface rises rapidly, leading to steel melting and ablation. These processes are modeled in the routine LEABLA. The melting occurs only at the surface, while the bulk of the cladding can remain below the melting point. To model this situation, the cladding is divided into two radial cells, with a thick inner cell and a thin cell at the outer surface (Figure 16.5.2). The thin cladding cell at the outer surface has a small thermal capacity and is thus quite sensitive to variations in the magnitude of the boundary heat fluxes. This cell is of key importance in the melting and ablation process. Ablation of the cladding can occur only if no fuel crust is present on cladding and:

$$T_{\text{cl,os,i}} > T_{\text{se,ablation}} \tag{16.5-26}$$

where

$$T_{\text{se,ablation}} = (T_{\text{se,so}} + T_{\text{se,}lq}) \cdot 0.5 \tag{16.5-27}$$

The fraction of the outer cell mass which is ablated is given by:

$$F_{\text{ablation,i}} = \frac{T_{\text{cl,os,i}} - T_{\text{se,ablation}}}{T_{\text{se,}lq} - T_{\text{se,ablation}}} \tag{16.5-28}$$

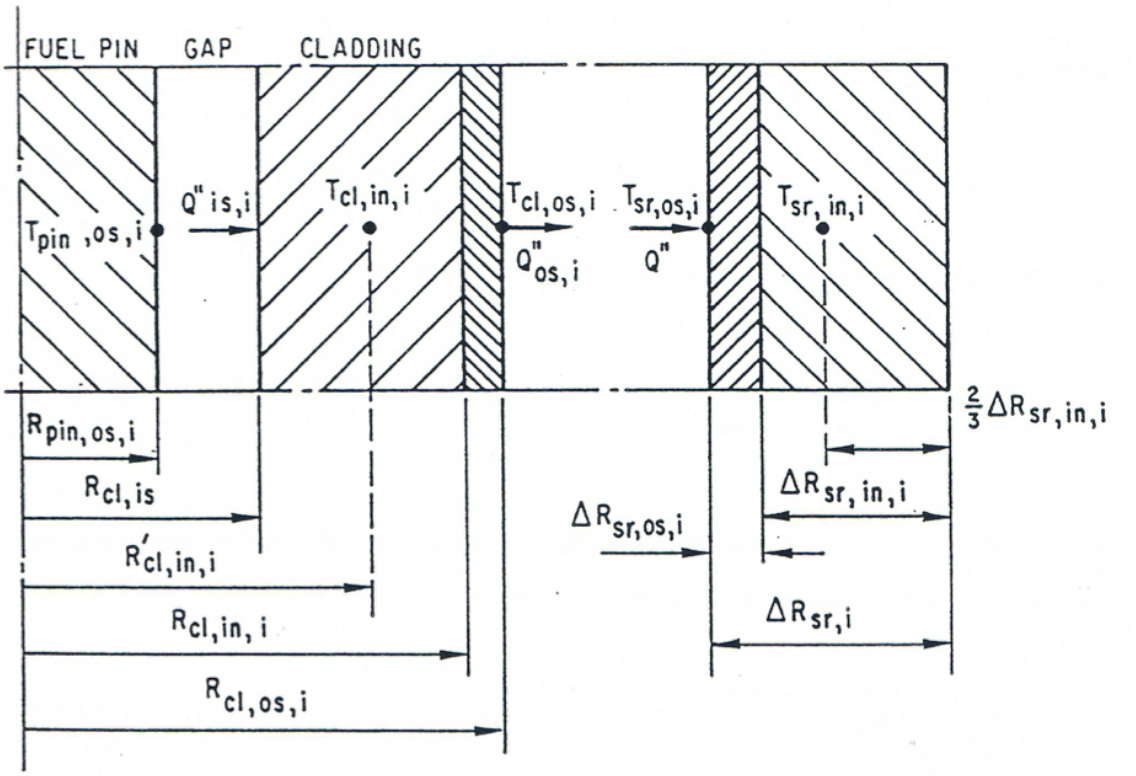


Figure 16.5.2: Temperature Grid in the Cladding and Structure

An amount $\Delta\rho'_{\text{se}}$ is removed from the outer clad cell and added to the molten steel field in the channel. The outer radius of the cladding is changed to:

$$R_{\text{cl,os,i}} = \sqrt{R^2_{\text{cl,in,i}} - \left(R^2_{\text{cl,os,i}} - R^2_{\text{cl,in,i}}\right) \cdot (1 - F_{\text{ablation,i}})} \tag{16.5-29}$$

Due to the explicit solution technique used in the cladding temperature calculation, a very thin outer cladding cell can lead to numerical instabilities in the temperature calculation or alternatively might require very small time steps. To avoid this difficulty while still maintaining an explicit method of solution in the temperature calculation, a lower limit ΔR_{min} is imposed on the thickness of the outer cladding cell. Whenever:

$$R_{\text{cl,in,i}} = R_{\text{cl,os,i}} - R_{\text{cl,in,i}} \leq \Delta R_{\text{min}} \tag{16.5-30}$$

The temperature calculation grid is restructured. The internal radius $R_{\text{cl,in,i}}$ is set to:

$$R_{\text{cl,in,i}} = R_{\text{cl,os,i}} - 2 \cdot \Delta R_{\text{min}} \tag{16.5-31}$$

and the temperature of the cladding cells is adjusted appropriately. This process continues until $R_{\text{cl,in,i}} = R_{\text{cl,is,i}}$, at which time the internal cladding cell disappears completely. The ablation process can still continue until the outer (and only) cladding cell reaches the thickness ΔR_{min}. Afterwards, no ablation is allowed until the remaining cladding reaches the melting point, when the cladding is completely removed and the fuel pin begins to transfer energy directly to the flowing components in the channel.

A similar procedure is used for the ablation of the hexcan wall. The ablation process is illustrated in Figure 16.5.1.

16.5.6 Fuel/Steel Chunk Melting

The moving solid chunks in the channel, both fuel and steel, can begin to melt due to heat transfer from the surrounding components and, in the case of the fuel chunks, due to internal heat generation. This process is modeled in the routine LELUME. The remelting of the fuel chunks occurs when:

$$h_{fl,\text{i}} > h_{fl,\text{melt}} \tag{16.5-32}$$

where

$$h_{fl,\text{melt}} = (h_{\text{fu,so}} + h_{\text{fu},lq}) \cdot 0.5 \tag{16.5-33}$$

The amount of molten fuel is obtained from:

$$\Delta\rho'_{fl,\text{i}} = \rho'_{fl,\text{i}} \cdot \frac{h_{fl,\text{i}} - h_{fl,\text{melt limit}}}{h_{\text{fu},lq} - h_{fl,\text{melt limit}}} \tag{16.5-34}$$

where

$$h_{fl,\text{melt limit}} = \frac{(h_{fl,\text{melt}} - 0.1 h_{\text{fu},lq})}{0.9} \tag{16.5-35}$$

The quantity $h_{ll,\text{melt limit}}$ is defined by Eq. (16.5-35) to be slightly below $h_{ll,\text{melt}}$. When melting occurs, as defined by Eq. (16.5-34), the enthalpy of the molten material is $h_{\text{fu},lq}$, and the enthalpy of he remaining chunks is $h_{ll,\text{melt limit}}$. Because $h_{ll,\text{melt limit}}$ is lower than $h_{ll,\text{melt}}$, which is used in Eq. (16.5-32) to trigger the melting process, the continuous melting of very small amounts of fuel is avoided. Melting of the fuel chunks in cell i will occur only when their enthalpy again reaches $h_{ll,\text{melt limit}}$. The size of the chunks is decreased appropriately, but their number remains unchanged. The molten fuel is added to the molten fuel in the channel:

$$\rho'_{\text{fu,i}} = \rho'_{\text{fu,i}} + \Delta\rho'_{\text{fu,i}} \tag{16.5-36}$$

and the energy and velocity of the molten fuel field are adjusted to reflect the addition of the molten fuel resulting from the remelting of the chunks.

A similar procedure is used for melting the steel chunks, which occurs when:

$$h_{sl,\text{i}} > h_{sl,\text{melt}} \tag{16.5-37}$$

where

$$h_{sl,\text{melt}} = (h_{\text{se,so}} + h_{\text{se},lq}) \cdot 0.5 \tag{16.5-38}$$

The amount of molten steel is obtained from

$$\Delta\rho'_{sl,\text{i}} = \rho'_{sl,\text{i}} \cdot \frac{h_{sl,\text{i}} - h_{sl,\text{melt limit}}}{h_{\text{se,lq}} - h_{sl,\text{melt limit}}} \tag{16.5-39}$$

16.5.7 Heat-transfer Calculation for the Steel Cladding and the Hexcan Wall

These calculations are performed in the routine LESOEN.

16.5.7.1 Heat-transfer Calculation for the Steel Cladding

The heat-transfer calculation for the steel cladding is performed using the two-node mesh illustrated in Figure 16.5.2. Two energy equations, one for each node, are solved explicitly. Due to ablation the geometry of the two cladding nodes can change in any time step.

The energy conservation equation for the outer cladding node in the axial cell i is:

$$\begin{aligned} \rho_{\text{se}} \cdot \frac{\partial \text{h}_{\text{cl,os,i}}}{\partial \text{t}} \cdot \left(R^2_{\text{cl,os,i}} - R^2_{\text{cl,in,i}}\right) \cdot \pi \cdot \Delta z_{\text{i}} \\ = k_{\text{se}} \cdot \frac{T_{\text{cl,in,i}} - T_{\text{cl,os,i}}}{R_{\text{cl,os,i}} - R'_{\text{cl,in,i}}} \cdot 2\pi R_{\text{cl,in,i}} \cdot \Delta z_{\text{i}} - \sum_{\text{j}} H_{\text{cl,j,i}} \\ \cdot \left(T_{\text{cl,os,i}} - T_{\text{j,i}}\right) \cdot \frac{A^L_{\text{cl,j,i}}}{N_{\text{pins}}} + Q_{\text{cl,j}} \cdot \frac{R_{\text{cl,os,i}} - R_{\text{cl,in,i}}}{\Delta R^0_{\text{cl}}} \end{aligned} \tag{16.5-40}$$

where the $\sum_{\text{j}}$ is performed over all the components in the channel that are in contact with the cladding and $Q_{\text{cl,i}}$ is the energy source in the cladding in cell i. The subscript se refers here to the solid steel, as opposed to the previous occurrences where it was followed by i (e.g. $\rho_{\text{se,i}}$), when it was referring to the molten steel in the channel. After integration over Δt, division by Δz_{i} and rearrangement, e.g., Eq. (16.5-40) becomes:

$$\begin{aligned} h^{n+1}_{\text{cl,os,i}} = h_{\text{cl,os,i}} + \frac{\Delta t}{\rho_{\text{se}} \cdot \pi \left(R^2_{\text{cl,os,i}} - R^2_{\text{cl,in,i}}\right)} \left\{ \frac{k_{\text{se}} \cdot 2\pi R_{\text{cl,in,i}}}{R_{\text{cl,os,i}} - R'_{\text{cl,in,i}}} \right. \\ \times \left(T_{\text{cl,in,i}} - T_{\text{cl,os,i}}\right) - \frac{\text{AXMX}}{N_{\text{pins}}} \left[H'_{\text{Na,cl,i}} \left(T_{\text{cl,os,i}} - T_{\text{Na,i}}\right)\right. \\ + H'_{\text{fu,cl,i}} \left(T_{\text{cl,os,i}} - T_{\text{fu,i}}\right) + H'_{\text{se,cl,i}} \left(T_{\text{cl,os,i}} - T_{\text{se,i}}\right) \\ + H'_{\text{cl,fi,i}} \left(T_{\text{cl,os,i}} - T_{\text{fl,i}}\right) + H'_{\text{cl,sl,i}} \left(T_{\text{cl,os,i}} - T_{\text{sl,i}}\right) \\ \left. \left. + H'_{\text{cl,ffc,i}} \left(T_{\text{cl,os,i}} - T_{\text{ffc,i}}\right)\right] + Q_{\text{cl,i}} \cdot \frac{R_{\text{cl,os,i}} - R_{\text{cl,in,i}}}{\Delta z_{\text{i}} \cdot \Delta R^o_{\text{cl}}} \right\} \end{aligned} \tag{16.5-41}$$

where $H'_{\text{j,cl,i}}$ are the generalized heat-transfer coefficients which have been defined previously.

The energy conservation equation for the inner cladding node in the axial cell is shown below:

$$\begin{aligned} \rho_{\text{se}} \frac{\partial \text{h}_{\text{cl,in,i}}}{\partial \text{t}} \cdot \left(R^2_{\text{cl,in,i}} - R^2_{\text{cl,is,i}}\right) \cdot \pi \cdot \Delta z_{\text{i}} \\ = -k_{\text{se}} \frac{T_{\text{cl,in,i}} - T_{\text{cl,os,i}}}{R_{\text{cl,os,i}} - R_{\text{cl,in,i}}} \cdot 2\pi R_{\text{cl,in,i}} \cdot \Delta z_{\text{i}} \\ + H_{\text{cl,in,pin,i}} \cdot 2\pi \cdot R_{\text{cl,is,i}} \cdot \Delta z_{\text{i}} \cdot \left(T_{\text{pin,os,i}} - T_{\text{cl,in,i}}\right) \\ + Q_{\text{cl,i}} \cdot \frac{R_{\text{cl,in,i}} - R_{\text{cl,is,i}}}{\Delta R^o_{\text{cl}}} , \end{aligned} \tag{16.5-42}$$

where $H_{\text{cl,in,pin,i}}$ is the heat-transfer coefficient between the inner cladding node and the outer fuel node. It takes into account the gap heat conductance and is defined as follows:

$$H_{\text{cl,in,pin,i}} = \frac{1}{\frac{1}{H_{\text{gap}}} + \frac{R'_{\text{cl,in,i}} - R_{\text{cl,is,i}}}{k_{\text{se}}}} \tag{16.5-43}$$

After integration over Δt and rearrangement, Eq. (16.5-42) becomes:

$$\begin{aligned} h^{n+1}_{\text{cl,in,i}} = h_{\text{cl,in,i}} - \frac{\Delta t}{\rho_{\text{se}} \cdot \pi \left(R^2_{\text{cl,in,i}} - R^2_{\text{cl,is,i}}\right)} \cdot \Bigg[& \frac{k_{\text{se}} \cdot 2\pi R_{\text{cl,in,i}}}{R_{\text{cl,os,i}} - R_{\text{cl,in,i}}} \\ & \cdot \left(T_{\text{cl,in,i}} - T_{\text{cl,os,i}}\right) - H_{\text{cl,in,pin,i}} \cdot 2\pi R_{\text{cl,is,i}} \\ & \cdot \left(T_{\text{pin,os,i}} - T_{\text{cl,in,i}}\right) - Q_{\text{cl,i}} \cdot \frac{R_{\text{cl,in,i}} - R_{\text{cl,is,i}}}{\Delta R^o_{\text{cl}} \cdot \Delta z_{\text{i}}} \Bigg] \end{aligned} \tag{16.5-44}$$

A special situation occurs whenever the inner cladding node disappears as a consequence of the ablation process, which has been described previously. This situation is indicated by setting the temperature $T_{\text{cl,in,i}}$, which is not longer used, to a negative arbitrary value, i.e., $T_{\text{cl,in,i}} = -100$. In this case, the outer cladding node exchanges heat directly with the fuel pin and the energy Eq. (16.5-41) is changed to:

$$\begin{aligned} h^{n+1}_{\text{cl,os,i}} = h'_{\text{cl,os,i}} + \frac{\Delta t}{\rho_{\text{se}} \cdot \pi \cdot \left(R^2_{\text{cl,os,i}} - R^2_{\text{cl,in,i}}\right)} \\ \cdot \Bigg[-\frac{\text{AXMX}}{N_{\text{pins}}} \cdot \sum_{\text{j}} H'_{\text{cl,j,i}} \left(T_{\text{cl,os,i}} - T_{\text{j,1}}\right) + H_{\text{cl,os,pin,i}} \cdot 2\pi R_{\text{cl,in,i}} \\ \cdot \left(T_{\text{pin,os,i}} - T_{\text{cl,os,i}}\right) + Q_{\text{cl,i}} \frac{R_{\text{cl,os,i}} - R_{\text{cl,in,i}}}{\Delta z_{\text{i}} \cdot \Delta R^o_{\text{cl}}} \Bigg] \end{aligned} \tag{16.5-45}$$

where $H_{\text{cl,os,pin,i}}$ is the heat-transfer coefficient between the outer cladding node and the outer pin node (when the inner node was vanished):

$$H_{\text{cl,os,pin,i}} = \frac{1}{\frac{1}{H_{\text{gap}}} + \frac{R_{\text{cl,os,i}} - R_{\text{cl,os,i}}}{k_{\text{se}}}} \tag{16.5-46}$$

The energy transferred between the cladding and the pin in each LEVITATE time step is integrated over the heat-transfer time step and stored under the name HFPICL (I). This quantity is then used in the pin heat-transfer calculation, in the routine PLHTR, as the pin boundary condition at $R = R_{\text{pin,os,i}}$.

Another special situation is the case when the cladding has been totally ablated, and the moving components in the channel are in direct contact with the fuel pin. For this case, the cladding temperature $T_{\text{cl,os,i}}$ is set equal to $T_{\text{pin,os,i}}$:

$$T_{\text{cl,os,i}} = T_{\text{pin,os,i}} \tag{16.5-47}$$

This allows all heat transfer to the channel components, as well as the freezing/melting processes, to be calculated correctly. Freezing of fuel on the bare fuel pin can still occur, if predicted by the freezing model. This situation can be identified in the output by the presence of undisrupted pin nodes, i.e., IDISR $(I) \neq 1$ with no cladding on them, i.e., WICLAD $(I) = 0$. In these cells, the output will indicate that no inner cladding node is present, i.e., $T_{\text{cl,in,i}} = -100$, and the temperature printed under $T_{\text{cl,os,i}}$ will represent the outer temperature of the fuel pin, according to Eq. (16.5-47).

The procedure outlined above is used in the fuel and blanket region of the fuel pin. A similar but simplified procedure is used for the remainder of the pin, where no fuel is present. A zero heat-flux boundary condition is used in these nodes.

The new cladding temperatures are finally obtained from the new enthalpies:

$$T^{n+1}_{\text{cl,os,i}} = T\left(h^{n+1}_{\text{cl,os,i}}\right); \quad T_{\text{cl,in,i}} = T\left(h^{n+1}_{\text{cl,in,i}}\right) \tag{16.5-48}$$

16.5.7.2 Heat-transfer Calculation for the Hexcan Wall

The heat-transfer calculation for the hexcan wall (the hexcan wall is alternatively referred to as structure) is performed using the two-cell mesh illustrated in Figure 16.5.2. *The outer structure cell is the cell facing the coolant channel*. This cell is fairly thin, and responds rapidly to changes in the heat-transfer from the coolant channel. This is necessary for the correct modeling of the structure ablation process, which has been described previously. It is assumed that no heat transfer occurs at the outer boundary of the hexcan wall, i.e., at the boundary facing the neighboring subassemblies. Because of this condition, and in order to improve the accuracy of the two-node calculation for the fairly thick structure, the temperature in the structure is assumed to have a parabolic variation:

$$T\left(\Delta R\right) = a \cdot \Delta R^2 + b \cdot \Delta R + c \tag{16.5-49}$$

where ΔR is the radial coordinate measured from the hexcan boundary facing the channel towards the pin. The coefficients a, b, and c are defined each time step by the conditions:

$$\frac{\mathrm{dT}}{\mathrm{d}\left(\mathrm{AR}\right)}\Big|_{\Delta \mathrm{R}} = -\Delta R_{\mathrm{sr,i}} = 0 \tag{16.5-50}$$

$$T\left(0\right) = T_{\mathrm{sr,os,i}} \tag{16.5-51}$$

$$T\left(-\Delta R_{\mathrm{sr,os,i}} - \frac{1}{3}\Delta R_{\mathrm{sr,in,i}}\right) = T_{\mathrm{sr,in,i}} \tag{16.5-52}$$

Using Eq. (16.5-50) through Eq. (16.5-52), it is found that the temperature gradient at the boundary between the outer and the inner structure nodes is given by:

$$\begin{aligned} &\frac{\mathrm{dT}}{\mathrm{d}\left(\Delta \mathrm{R}\right)}\Big|_{\Delta \mathrm{R}=-\Delta R_{\mathrm{sr,os,i}}} \\ &= \left(T_{\mathrm{sr,os,i}} - T_{\mathrm{sr,in,i}}\right) \cdot \frac{18 \cdot \Delta R_{\mathrm{sr,in,i}}}{5\Delta R_{\mathrm{sr,i}}^2 + 8\Delta R_{\mathrm{sr,i}} \cdot \Delta R_{\mathrm{sr,os,i}} - 4\Delta R_{\mathrm{sr,os,i}}^2} \end{aligned} \tag{16.5-53}$$

This temperature gradient expression will be used in the energy-conservation equations for the structure. The equation for the outer structure cell is written:

$$\begin{aligned} &\rho_{\mathrm{se}} \cdot \frac{\partial \mathrm{h}_{\mathrm{sr,os,i}}}{\partial \mathrm{t}} \cdot \Delta R_{\mathrm{sr,os,i}} \cdot \Delta z_{\mathrm{i}} \cdot L_{\mathrm{sr,i}} \\ &= -k_{\mathrm{se}} \cdot \frac{\partial \mathrm{T}}{\partial\left(\Delta \mathrm{R}\right)}\Big|_{\Delta R=-\Delta R_{\mathrm{sr,os,i}}} \cdot \Delta z_{\mathrm{i}} \cdot L_{\mathrm{sr,i}} \\ &\quad + \sum_{\mathrm{j}} H_{\mathrm{sr,j,i}} \cdot \left(T_{\mathrm{j,i}} - T_{\mathrm{sr,os,i}}\right) \cdot A_{\mathrm{sr,j,i}}^{L} \end{aligned} \tag{16.5-54}$$

where $\sum_{\mathrm{j}}$ is performed over all the components in the channel that are in contact with the structure. Although ΔR is used as the "radial" coordinate, the hexcan wall is in fact assumed to be flat, with thickness ΔR_{i} and perimeter $L_{\mathrm{sr,i}}$.

After integration over Δt, rearrangement and use of Eq. (16.5-53), the above equation becomes:

$$\begin{aligned} h_{\mathrm{sr,os,i}}^{n+1} = h_{\mathrm{sr,os,i}} + \frac{\Delta t}{\rho_{\mathrm{se}} \cdot R_{\mathrm{sr,os,i}} \cdot L'_{\mathrm{sr,i}}} \cdot \Bigg[&- \frac{k_{\mathrm{se}} L'_{\mathrm{sr,i}} \cdot 18 R_{\mathrm{sr,in,i}}}{5R_{\mathrm{sr,i}}^2 + 8R_{\mathrm{sr,i}} \cdot R_{\mathrm{sr,os,i}} - 4R_{\mathrm{sr,os,i}}^2} \cdot \left(T_{\mathrm{sr,os,i}} - T_{\mathrm{sr,in,i}}\right) \\ &+ H'_{\mathrm{Na,sr,i}} \cdot \left(T_{\mathrm{Na,i}} - T_{\mathrm{sr,os,i}}\right) + H'_{\mathrm{fu,sr,i}} \cdot \left(T_{\mathrm{fu,i}} - T_{\mathrm{sr,os,i}}\right) + H'_{\mathrm{se,sr,i}} \cdot \left(T_{\mathrm{se,i}} - T_{\mathrm{sr,os,i}}\right) \\ &+ H'_{\mathrm{fl,sr,i}} \cdot \left(T_{\mathrm{fl,i}} - T_{\mathrm{sr,os,i}}\right) + H'_{sl,\mathrm{sr,i}} \cdot \left(T_{sl,\mathrm{i}} - T_{\mathrm{sr,os,i}}\right) + H'_{\mathrm{ffs,sr,i}} \cdot \left(T_{\mathrm{ffs,i}} - T_{\mathrm{sr,os,i}}\right) \Bigg] \end{aligned} \tag{16.5-55}$$

where

$$L'_{\mathrm{sr,i}} = \frac{L_{\mathrm{sr,i}}}{\mathrm{AXMX}} \tag{16.5-56}$$

and $H'_{\mathrm{j,sr,i}}$ are the generalized heat-transfer coefficients which were defined previously as follows:

$$H'_{\mathrm{j,sr,i}} = H_{\mathrm{j,sr,i}} \cdot \frac{A_{\mathrm{j,sr,i}}}{\mathrm{AXMX} \cdot \Delta z_{\mathrm{i}}} \tag{16.5-57}$$

The new temperature of the structure outer node is then obtained from its enthalpy:

$$T^{n+1}_{\mathrm{sr,os,i}} = T\left(h^{n+1}_{\mathrm{sr,os,i}}\right) \tag{16.5-58}$$

The energy equation for the inner structure cell (i.e., the cell which is not in contact with the coolant channel) is written as follows:

$$\begin{aligned} \rho_{\mathrm{se}} \cdot \frac{\partial \mathrm{h}_{\mathrm{sr,in,i}}}{\partial \mathrm{t}} \cdot \Delta R_{\mathrm{sr,in,i}} \cdot \Delta z_{\mathrm{i}} \cdot L_{\mathrm{sr,i}} \\ = k_{\mathrm{se}} \frac{\partial \mathrm{T}}{\partial\left(\Delta \mathrm{R}\right)}\Big|_{\Delta \mathrm{R}=-\Delta R_{\mathrm{sr,os,i}}} \cdot \Delta z_{\mathrm{i}} \cdot L_{\mathrm{sr,i}} \end{aligned} \tag{16.5-59}$$

After integration over Δt and rearrangement, the above equation becomes:

$$\begin{aligned} h^{n+1}_{\mathrm{sr,in,i}} = h_{\mathrm{sr,in,i}} \\ + \frac{\Delta t \cdot k_{\mathrm{se}} \cdot 18 \Delta R_{\mathrm{sr,in,i}} \cdot \left(T_{\mathrm{sr,in,i}}\right)}{\rho_{\mathrm{se}} \cdot \Delta R_{\mathrm{sr,in,i}} \cdot \left(5\Delta R^2_{\mathrm{sr,i}} + 8 \cdot \Delta R_{\mathrm{sr,i}} \cdot \Delta R_{\mathrm{sr,os,i}} - 4\Delta R^2_{\mathrm{sr,os,i}}\right)} \end{aligned} \tag{16.5-60}$$

The new temperature of the structure node is then obtained from its enthalpy:

$$T^{n+1}_{\mathrm{sr,in,i}} = T\left(h^{n+1}_{\mathrm{sr,in,i}}\right) \tag{16.5-61}$$

A special situation occurs when the "inner" structure cell has disappeared completely as a result of the ablation process. This situation is indicated by setting $T_{\mathrm{sr,in,i}}$ to an arbitrary negative value, i.e., $T_{\mathrm{sr,in,i}} = -100$. In this case, only the Eq. (16.5-54) for the outer structure cell is solved in a simplified form. The term $k_{\mathrm{se}} \frac{\partial \mathrm{T}}{\partial(\Delta R)}\Big|_{\Delta R=-\Delta R_{\mathrm{sr,os,i}}}$ is set to zero, and the final equation is shown below:

$$h^{n+1}_{\mathrm{sr,os,i}} = h_{\mathrm{sr,os,i}} + \frac{\Delta t}{\rho_{\mathrm{se}} \cdot \Delta R_{\mathrm{sr,os,i}} \cdot L'_{\mathrm{sr,i}}} \cdot \sum_{\mathrm{j}} H'_{\mathrm{j,sr,i}} \cdot \left(T_{\mathrm{j,i}} - T_{\mathrm{sr,os,i}}\right) \tag{16.5-62}$$

When the "inner" structure cell has been removed and the outer cell has reached both a minimum thickness and the melting point, the hexcan wall is assumed to be breached. Intersubassembly fuel motion is likely to begin. Theoretically, this should be the end of the LEVITATE calculation and the beginning of a transition-phase calculation. However, the code will only print a warning message and the calculation will continue assuming that the hexcan wall will maintain the minimum thickness, although its temperature has risen above the melting range.

16.6 LEVITATE Interaction with Other Modules

16.6.1 Interaction with the Point Kinetics Module

The LEVITATE interface with the point kinetics model has been described briefly in Section 16.1.3.3.3.

The axial mass and temperature distributions needed to calculate the reactivity feedbacks in the point kinetics module are calculated in the LEVITATE driver routine, LEVDRV, at the end of each PRIMAR time step. The changes in reactivity are due to changes in the distribution of sodium, fuel and steel in the active core region, and changes in the fuel temperature. The formulation for these feedbacks is given in Chapter 4, Section 4.5.

16.6.2 Coupling with the Primary Loop Module

As explained briefly in Section 16.1.3.3.1, LEVITATE can be coupled with either PRIMAR-1 or the more advanced PRIMAR-4 module. The PRIMAR-1 model is used whenever the input variable IPRION has a value not equal to 4. In this case, LEVITATE uses the constant outlet plenum pressure PX which is input and an inlet plenum pressure which is determined by PRIMAR-1. This is the only information needed in LEVITATE to calculate the motion of the liquid sodium slugs. A temperature calculation within the sodium slugs is not currently performed in LEVITATE and thus no plenum temperature information is expected. If the PRIMAR-1 option has been chosen, LEVITATE will not return any information to the primary loop module.

When the PRIMAR-4 option is selected (IPRION=4), LEVITATE will use the time-dependent inlet and outlet pressures which are calculated by PRIMAR-4 to determine the dynamics of the liquid sodium slugs. The inlet pressure used in each LEVITATE time step is calculated as follows:

$$P_{\text{inlet}}(t) = P_{\text{inlet}}(t_{\text{PR1}}) + (t - t_{\text{PR1}}) \cdot \frac{\partial \text{P}_{\text{inlet}}}{\partial \text{t}} \tag{16.6-1}$$

where

$P_{\text{inlet}}(t_{\text{PR1}})$ - is the inlet pressure calculated by PRIMAR-4 at the beginning of the current primary loop time step.

$\frac{\partial \text{P}_{\text{inlet}}}{\partial \text{t}}$ - is the rate of change of the inlet pressure during the previous primary loop time step, calculated by PRIMAR-4

t_{PR1} - is the time at the beginning of the current primary loop time step.

The outlet pressure is calculated in the same manner:

$$P_{\text{outlet}}(t) = P_{\text{outlet}}(t_{\text{PR1}}) + (t - t_{\text{PR1}}) \cdot \frac{\partial \text{P}_{\text{outlet}}}{\partial \text{t}} \tag{16.6-2}$$

When the PRIMAR-4 option is used, LEVITATE provides PRIMAR-4 with the total sodium masses ejected into or received from the plena during a primary loop time step:

$$\Delta M_{\text{Na,ic,inlet}} = N_{\text{subas,ic}} \cdot \int_{t_{\text{PR1}}}^{t_{\text{PR2}}} W_{\text{Na,inlet}} \cdot \text{dt} \tag{16.6-3}$$

and

$$\Delta M_{\text{Na,ic,outlet}} = N_{\text{subas,ic}} \cdot \int_{t_{\text{PR1}}}^{t_{\text{PR2}}} W_{\text{Na,outlet}} \cdot \text{dt} \tag{16.6-4}$$

where

t_{PR1} - is the time at the beginning of the PRIMAR-4 time step

t_{PR2} - is the time at the end of the PRIMAR-4 time step

W_{Na} - is the sodium mass flow rate in kg/s

LEVITATE also provides PRIMAR-4 with the channel mass flow rates at the end of the primary loop time step. As long as single phase liquid sodium is ejected into the upper or lower plena, temporal integrals of the product (sodium flow rate $\times$ temperature) are also provided by LEVITATE. However, when the upper liquid sodium slug has been ejected out of the subassembly, LEVITATE calculates the total enthalpy of the two-phase mixture ejected during a primary loop time step:

$$\begin{aligned} \Delta h_{\text{ic}} = [(1 - X_{\text{Na,IITP-1}}) \cdot h_{\text{Nl}} + X_{\text{Na,IITP-1}} \cdot h_{\text{Nv}}] \\ \cdot W_{\text{Na}} \cdot N_{\text{subs,ic}} \cdot (t_{\text{PR2}} - t_{\text{PR1}}) \end{aligned} \tag{16.6-5}$$

where

$X_{\text{Na,IITP-1}}$ - is the sodium quality in the axial node IITP-1, the highest full node in the subassembly.

h_{N1}, h_{Nv} - are the enthalpies of the liquid sodium and sodium vapor, respectively, in the axial node IITP-1.

16.7 Detailed Logic Flow Description

This section describes in detail the logical sequence of the solution method used in LEVITATE. The structure of the LEVITATE driver as well as initiation and interfacing considerations have been presented in Section 16.1.3. Some information about the interaction between LEVITATE models was presented in Section 16.1.2.4 and the method of solution for the coolant channel model was described in Section 16.4.2. This section will use, and occasionally repeat some of this information in order to present a comprehensive picture of the general solution method. Some considerations about the data management and time-step selection are also presented.

16.7.1 Data Management Considerations

Every time LEVITATE begins calculations in a channel, the permanently stored information is retrieved from the data container and loaded in the common blocks. This operation is performed in the TSTHRM module. A number of arrays and other data, however, which can be calculated from the permanently stored variables, are not stored in the permanent storage. These data are recalculated in the routine LESET2 every time LEVITATE receives control in a certain channel, i.e., at the beginning of a primary time step. These variables are stored in temporary common blocks, and kept only as long as LEVITATE retains control in the channel. At the end of the primary time step, when LEVITATE returns control to TSTHRM, these variables are lost. Only the permanent common blocks are saved in the data container. This procedure is used in order to reduce the amount of storage required for SAS4A.

16.7.2 Logic Flow for Solution Advancement

As explained in Section 16.1.2, the LEVITATE model can be viewed as being made up of three large models, i.e, the coolant channel hydrodynamic model, the in-pin fuel motion model and the freezing/melting model describing the fuel pins and the hexcan wall. During each time step, LEVITATE advances the solution for these models by calculating the value of all variables at the end of the time step. The time-step selection is described in Section 16.7.3. The simplified modular chart of the LEVITATE model is shown again in Figure 16.7.1.

The LEVITATE calculation begins by advancing the solution for the channel hydrodynamic model. The changes in the position of material boundaries are calculated first. The new interface positions, at the end of the time step, are calculated for all components, except sodium in the LEIF (LEVITATE INTERFACE) routine. The position of the sodium slugs, which determine the boundaries of the sodium region, is calculated in the LEREZO **LE**VITATE **REZON**ING) routine. This routine can add (or remove) nodes to the LEVITATE compressible region as the sodium slugs move out of (or remove) nodes to the LEVITATE compressible region as the sodium slugs move out of (or into) the channel.

The mass conservation equation is solved next for all components and all axial locations. Each equation is solved explicitly, i.e., the convective fluxes are based on the generalized densities present in each cell at the beginning of the time step. These calculations are performed in the LEMACO (**LE**VITATE **MA**SS **CO**NSERVATION) routine.

The LEVOFR (**LE**VITATE **VO**LUME **FR**ACTION) routine uses the new densities to determine the volume fraction of each component at each axial location. Using these volume fractions, the LEVOFR routine also determines the flow regime in each axial cell. This flow regime will be assumed to exist in the cell of the

duration of the time step. It is worth noting that this is an "implicit type" assumption as the flow regimes are based on the densities calculated at the end of the time step.

The next routine called is LEGEOM (**LE**VITATE **GEOM**ETRY) which determines the geometrical characteristics defining each local configuration. This routine will change the thickness of the fuel crust as necessary or determine

the fraction of the cladding circumference covered by the liquid fuel in the partial annular flow regime. In general, LEGEOM calculates the area of contact between various components in various flow regimes. These areas will be used later in calculating the energy and momentum transfer between various components.

The next routine called is LETRAN (**LE**VITATE **TRAN**SFER) which calculates the heat-transfer and friction coefficients for all axial locations and among all components that are in direct contact. Thus, the code will use the flow regimes present at a certain location to determine which heat-transfer and friction coefficients have to be calculated. The allowable exchanges for each flow regime and each component have already been presented in Figure 16.4.3 through Figure 16.4.10. Once contact areas and the corresponding heat-transfer coefficients have been calculated, the energy conservation equation can be solved for all components.

The LESOEN (**LE**VITATE **SO**LID, LIQUID AND STATIONARY **EN**ERGY EQUATION) routine is called to solve the energy conservation equations for the fuel and steel channels, liquid fuel, liquid steel, stationary cladding, hexcan wall, frozen fuel on the cladding and frozen fuel on the hexcan wall. All equations are solved explicitly, i.e., the convective fluxes are based on beginning of time-step densities, thus allowing the axial decoupling of the equations. The energy equation for sodium (two-phase or single-phase vapor) and fission gas is solved in LENAEN (**LE**VITATE SODIUM-**NA** **EN**ERGY). The energy equations for fuel and steel vapor are solved LEFUVA (**LE**VITATE **FU**EL **VA**POR ENERGY) and LESEVA (**LE**VITATE **FU**EL **VA**POR ENERGY) and LESEVA (**LE**VITATE **S**TEEL **VA**POR ENERGY), respectively. The new temperatures calculated in LENAEN, LEFUVA and LESEVA are used to determine the new pressure of each of the compressible components and thus the total new pressure.

At this point, the only variables that remain to be calculated in order to complete the solution for the coolant channel hydrodynamic model are the velocities of various components. However, before calculating the new velocities, the hydrodynamic in-pin model is used to advance the in-pin solution in the LE1PIN and LE2PIN routines. Due to the strategy, the changes n the channel hydrodynamics due to the interaction with the in-pin motion will be taken into account in the velocity calculation. The routine LE1PIN solves the mass and energy conservation equations for the in-pin hydrodynamic model. It also calculates the enlargement of the pin cavity and the amount of molten fuel added to the cavity, as well as the amount of fuel ejected from the cavity via the cladding rip and/or the ends of the fuel-pin stubs. The LE2PIN routine completes the solution of the in-pin hydrodynamic model by solving the momentum conservation equations for all the cells in the cavity and determining the new in-pin velocities at the end of the time step. This routine determines the next maximum time step acceptable for the in-pin hydrodynamic model, DTPIN which will then be used in the LEVIATE driver as explained in Section 16.7.3. The in-pin model interacts directly with the channel model via the fuel injection process, which, as mentioned above, is described in the LE1PIN routine. Molten fuel and fission gas are ejected from cavity into the channel, leading to changes in the local pressure.

The solution of the channel hydrodynamic model is then completed by solving the momentum equation for each of the three velocity fields (fuel/steel, gas mixture and solid chunks) in the routine LEMOCO (**LE**VITATE **MO**MENTUM **CO**NSERVATION). The method of solution is still explicit and the equations are uncoupled axially, but the equations for all three fields are solved simultaneously rather than independently, as was done in the mass and energy equations. Also it is important to note that the pressures used in the momentum equation are the pressures at then end of the time step. The LEMOCO routine also calculates the new velocities of the lower and upper liquid sodium slugs, which will be used in the next time step in the LEREZO routine. It also calculates the quantities which are fed back to the primary loop model as explained in Section 16.1.3.3.1 and Section 16.6.2. The sodium slugs provide the axial boundaries for the interactive region modeled by LEVITATE and their molten can affect significantly the fuel motion in the coolant channel.

All the routine called after LEMOCO describe melting or freezing processes related to the solid chunks, cladding

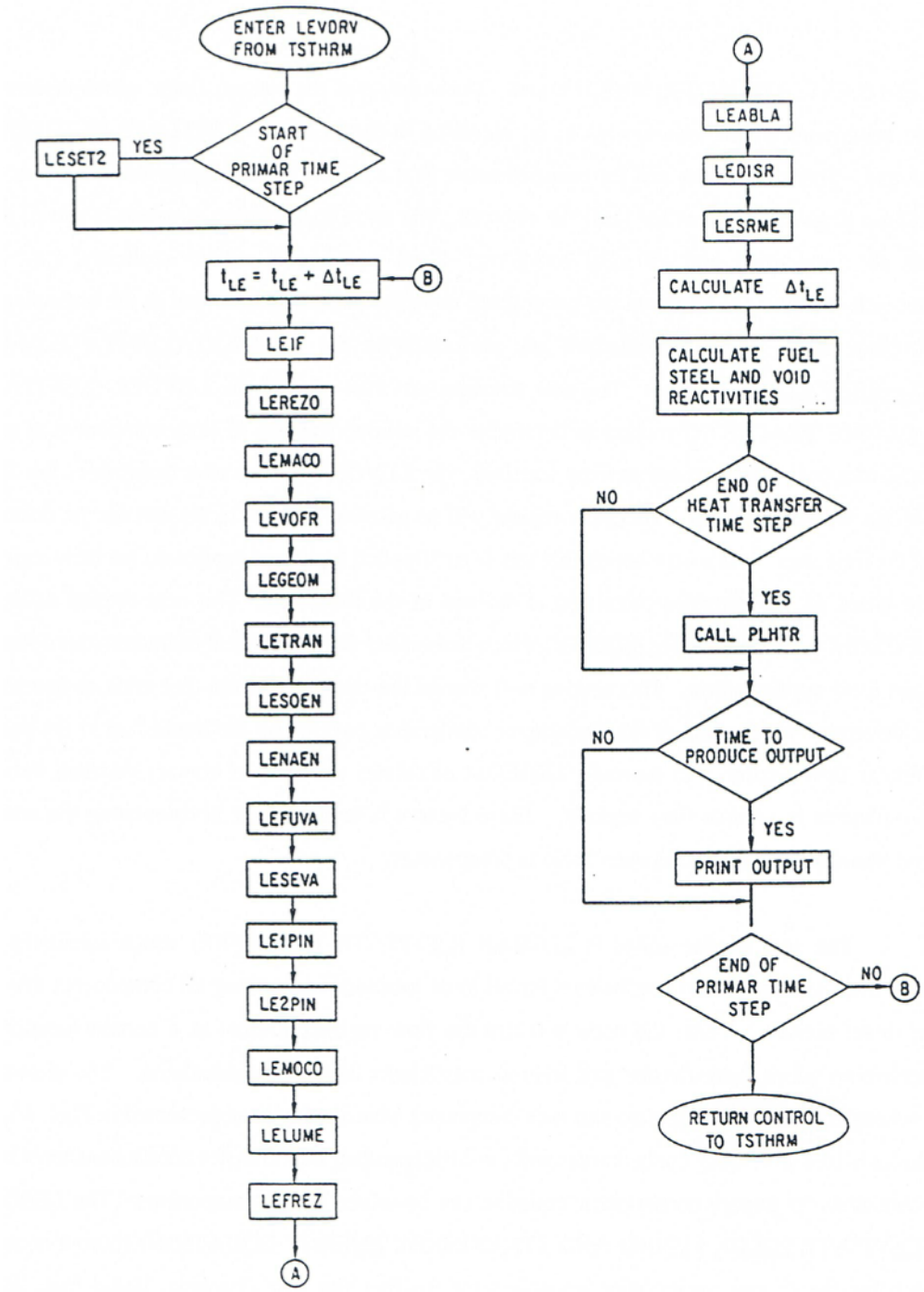


Figure 16.7.1: Simplified Modular Chart of the LEVITATE Module

or the hexcan wall. The routine LELUME (**LE**VITATE CHUNK-**LU** **MEL**TING) calculates the melting and the size changes of the solid fuel/steel chunks at all axial locations. The routine LEFREZ (**LE**VITATE **FREEZ**ING AND MELTING) then models a series of important processes, such as fuel/steel freezing and crust formation, fuel/steel chunk formation, fuel crust remelting and breakup. Next called is the routine LEABLA (**LE**VITATE **ABLA**TION) which calculates the gradual melting and ablation of the cladding and hexcan wall. The routine LEDISR (**LE**VITATE **DISTR**UPTION) performs the disruption of the fuel pin, whenever a node disruption is predicted. The disruption, which can occur in one or more nodes in any time step, leads to changes in geometry, mass, energy and pressure various components present in the respective cell. Finally, the routine LESRME (**LE**VITATE **S**T**R**UCTURE **ME**LTING) calculates the rupture of the hexcan wall, due to melting and/or pressure burst effects.

Using the material distributions at the end of the time step, LEVITATE then calculates the fuel, steel and sodium mass distributions, which, at the end of the primary time step, are fed back to the point kinetics module, as explained in Section 16.1.3.3.3 and Section 16.6.1. Finally, if the end of a heat-transfer time step is reached, LEVITATE class the routine PLHTR, which calculates the new temperatures in the solid fuel pin at all axial locations. These temperatures will be used in the next time step in the LE1PIN routine to calculate the new cavity diameter and the heat flux between the outer fuel pin surface and the cladding.

16.7.3 Time-step Considerations

The LEVITATE driver routine, LEVDRV can be called by the SAS4A transient driver TSTHRM at any time during a primary loop time step. The primary time step is common for all SAS4A calculational channels and is always smaller than or equal to the main time step used in the point kinetics calculation.

When the fuel-pin failure is detected in a given channel, the LEVITTE and PLUTO2 time, TIMEPL, is set to zero in the FAILUR routine. Then LEVITATE is entered directly after pin failure, TIMEPL is initially advanced by adding the LEVITATE minimum time step DTPLIN to the time TIMEPL. However, if LEVITATE is entered after PLUTO2 has been active in a given channel, the TIMEPL will be incremented by the last time step calculated by PLUTO2, as explained in Section 14.7.2. Subsequently, the time TIMEPL will be advanced by the time step calculated by LEVITATE< as described below.

A maximum time step DTPIN, is calculated first for the in-pin hydrodynamic model, as outlined in Section 14.2.8. Then, once all LEVITATE calculations for the current time step are completed, a new time step for the channel hydrodynamic model is calculated in the LEVDRV routine. The details of this calculation are presented in Section 16.4.3.11. This time step is then compared with the in-pin time step and with the present maximum value $2 \cdot 10^{-4}$ s. The smallest value is retained as the LEVITATE time step. This value is further compared with the input minimum time step, DTPLIN and the larger value is retained. Finally, the LEVITATE time step can be cut back if the newly determined time:

$$t_{\mathrm{LE}}^{n+1} = t_{\mathrm{LE}}^{n} + \Delta t \tag{16.7-1}$$

extends beyond the end of a heat-transfer time step. In this case, the new LEVITATE time step will be reduced so that the new LEVITATE time t_{LE}^{n+1} will coincide with the end of the heat-transfer time step.

As shown in the flow diagram 16.7-1, LEVITATE retains control and advances the solution a given channel until the end of a primary loop time step is reached.

16.7.4 List of LEVITATE Subroutines

This section contains a list of all the subroutines that are part of the LEVITATE model. These routines are listed in the order they are called:

Name	Description
LEVDRV	LEVITATE driver routine.
LEIF	Calculates the new boundary region locations for all the material components, except sodium.
LEREZO	Calculates the new boundary locations for the sodium region, as determined by the dynamics of the liquid sodium slugs.
LEMACO	Solves the conservation equations for all the components present in the coolant channel.
LEVOFR	Calculates the volume fraction occupied by each channel component at all axial locations. It also determines the flow regime present in the channel at all axial locations.
LEGEOM	Determines the local geometry, taking into account the previous configuration and the current flow regime.
LETRAN	Calculates the heat and momentum transfer coefficients between various components.
LESOEN	Solves the coolant channel energy conservation equations for liquid fuel and steel, chunks, fuel crusts, cladding and hexcan wall.
LENAEN	Solves the coolant channel energy conservation equation for the sodium and fission-gas mixture in the coolant channel.
LEFUVA	Solves the energy conservation equation for the fuel vapor in the channel.
LESEVA	Solves the energy conservation equation for the steel vapor in the channel.
LE1PIN	Solves the mass and energy conservation equation for the in-pin hydrodynamic model. It also calculates the ejection of molten fuel from the cavity into the coolant channel.
LE2PIN	Solves the momentum conservation equations for the in-pin hydrodynamic model. Determines the next maximum time step for the in-pin model.
LEMOCO	Solves the momentum conservation equations for the channel hydrodynamic model. Also solves the momentum equations for the liquid sodium slugs.
LELUME	Examines the melting of the fuel and steel chunks. When melting occurs, performs al the required operations.
LEFREZ	Models the freezing of fuel and steel in the channel, crust formation, crust remelting, and heat-up. This routine is used when the chunk model is not active (ICHUCK=O).
LEFRLU	Models the freezing of fuel and steel in the channel, crust and chunk formation, crust remelting and breakup. This routine is used instead of LEFREZ when the chunk model is used (ICHUNK=1).
LEABLA	Models the ablation of cladding and/or hexcan wall, whenever necessary.
LEDISR	Models the disruption of the fuel pins at any axial location.

Other routines called in LEVITATE, which are shared with other modules, are:

Name	Description
PLHTR	Calculates the transient heat transfer in the solid fuel pin. It is called from the LEVITATE driver. This routine can also be called from eh PLUTO2 driver routine.
HHGAP	Determines the heat-transfer coefficient between the fuel pin and cladding. This routine is called from LESOEN. It can also be called from PLUTO2 or DEFORM.
UTS	Determines the ultimate tensile strength of the cladding. It is called from LEIF in order to determine the cladding rip propagation.
EGFUTE	Provides the fuel enthalpy as a function of temperature. It is used frequently in LEVITATE and PLUTO2.
TEFUEG	Provides the fuel temperature as a function of enthalpy.
EGSETE	Provides the steel enthalpy as a function of temperature.
TESEEG	Provides the steel temperature as a function of enthalpy.

16.8 Input Parameters Relevant to LEVITATE

The input parameters relevant to LEVITATE are summarized in Table 16.8.1. A description of these parameters can be found in the SAS4A input listing attached to this document. Table 16.8.1 lists the recommended values for these parameters and the sections and equations were those parameters are mentioned in the text. This list of equations is not necessarily exhaustive, and some input parameters might appear in other equations, in addition to those listed in the table (e.g., AXMX appears in many places and it was not possible to list all occurrences). Comments have been also added occasionally to complement the description given in the SAS4A input listing.

Table 16.8.1: Input Description

Input Location	FORTRAN Variable	Symbol	Section Reference	Equation Reference	Suggested Value (MKS)	Comments
Block 1, INPCOM						
37	KFAILP		14.3.3		0	Controls the axial pin failure propagation calculation.
38	NCPLEV		14.3.4 *14.4.3.2* *16.1.3.1*		3	Switch from PLUTO2 to LEVITATE when NCPLEV axial clad nodes have exceeded the cladding liquidus temperature.
39	NFUELD		16.1.3.4		−5	Number of dollars of fuel reactivity which has to be reached to terminate the calculation.
42	NSLEEX		16.1.3.4		10	Number of fully molten hexcan cells in a subassembly which has to be reached to terminate the calculation.

continues on next page

Table 16.8.1 – continued from previous page

Input Location	FORTRAN Variable	Symbol	Section Reference	Equation Reference	Suggested Value (MKS)	Comments
44	INRAEJ		16.3.1		1	If INRAEJ=1 the ejection of in-pin fuel is calculated using a mechanistic model. If 0 the parametric fuel ejection calculation is used (see **CIPINJ**).
47	ICHUNK				0	If ICHUNK=1 the chunk model is operational. Although the physical picture is more complete when ICHUNK-1, the chunk model has not yet been thoroughly validated. When ICHUNK=0 the chunk model is disabled.
48	ILUBLK				0	This variable is relevant only when the chunk model is active i.e. ICHUNK=1. If ILUBLK=1 the bulk fuel freezing leads to chunk formation when no solid support for crust formation is available. If ILUBLK=0 no chunk formation occurs under the circumstances mentioned above.
Block 13, PMATCM						
1126	CIBBIN		16.4.3.9.1		0.7	
1127	CIREFN		16.4.3.8.2	Eq. (16.4-198)	2100	
1128	CIFRFU		16.4.3.8.2	Eq. (16.4-198)	0.03	
1129	CIFUMO		16.4.3.8.2	Eq. (16.4-206)	1	Some of the in-pin axial momentum might be lost during the ejection process, due to interactions with the stationary components.

continues on next page

Table 16.8.1 – continued from previous page

Input Location	FORTRAN Variable	Symbol	Section Reference	Equation Reference	Suggested Value (MKS)	Comments
1133	CIA3		16.4.3.10.1. *16.4.3.10.1.* *16.4.3.10.1.*	Eq. (16.4-310) Eq. (16.4-314) Eq. (16.4-319)		
1135	CIA5		16.4.3.8.1	Eq. (16.4-173)	-2.7	
1136	CIA6		16.4.3.8.1	Eq. (16.4-170) Eq. (16.4-172)	0.1068	
1139	CPFU	$c_{\mathrm{p,fu}}$	16.4.3.10.1.	Eq. (16.4-284) Eq. (16.4-286)	500.0	
1140	CDFL	k_{fu}	16.4.3.10.1. *16.4.3.10.1.*	Eq. (16.4-289) Eq. (16.4-290) Eq. (16.4-306)	3.0	
1141	CMNL	$C_{\Delta \mathrm{P,Na}}$	16.4.3.5	Eq. (16.4-94) Eq. (16.4-95) Eq. (16.4-96)	$4.6 \cdot 10^{-10}$	
1142	CDNL	k_{nl}	16.4.3.10.1.	Eq. (16.4-343)	50	
1144	CDVG	k_{Mi}	16.4.3.10.1.	Eq. (16.4-344)	0.067	
1146	CFNACN		16.4.3.10.1.	Eq. (16.4-345)	$6.0 \cdot 10^{4}$	
1147	CFNAEV		16.4.3.10.1.	Eq. (16.4-344)	$6.0 \cdot 10^{5}$	
1148	FIFNGB			Eq. (14.2-18)	0.1	
1150	VIVG	μ_{vg}	16.4.3.8.1	Eq. (14.2-18)	0.1	
1151	EGFUSO	$H_{\mathrm{fu,so}}$	16.5.3.1	Eq. (16.5-2)	$1.0 \cdot 10^{6}$	
1155	C1VIPR				$3.0 \cdot 10^{-3}$	
1157	SUFU	σ_{fu}	16.4.3.8.1	Eq. (16.4-170)	0.45	

continues on next page

Table 16.8.1 – continued from previous page

Input Location	FORTRAN Variable	Symbol	Section Reference	Equation Reference	Suggested Value (MKS)	Comments
1161	EGBBLY	$H_{\text{fu,freeze}}$	16.5.3.1	Eq. (16.5-1)		Fuel freezing is initiated when the fuel enthalpy drops below EGBBLY should be between EGFUSO and EGFULQ.
1162	VIFULQ	μ_{fu}	16.4.3.10.1.	Eq. (16.4-284)		
1164	DTPLIN		16.7.3		$2.5 \cdot 10^{-5}$	
1165	AXMX		16.4.2	Eq. (16.4-1) etc.		Cross sectional area of subassembly, limited by the outside perimeter
1166	EPCH		14.3.2	Eq. (14.2-1)	1	
1168	DTPLP		16.7			The frequency of the LEVITATE output should be selected by the user.
1169	FNMELT			Eq. (14.2-2) Eq. (14.2-3)	0.9	
1170	CIRTFS			Eq. (14.2-21)	16.67	
1174	CIANIN		16.4.3.9.2		0.5	
1175	TEFAIL			Eq. (14.3-11)	TESOL(1)	
1176	FNARME			Eq. (14.3-10)	**FSPEC**	FNARME should be consistent with FSPEC when the input controlled failure propagation is used. However, a compromise might be necessary because FSPEC is channel dependent while FNARME is not.
1177	PRFAIL			Eq. (14.3-13)	0	
1188	CMFU			Eq. (14.2-40)	$6.0 \cdot 10^{-11}$	
1195	CDCL	k_{se}		Eq. (16.5-41) Eq. (16.5-42)	32.0	

continues on next page

Table 16.8.1 – continued from previous page

Input Location	FORTRAN Variable	Symbol	Section Reference	Equation Reference	Suggested Value (MKS)	Comments
1198	RHSLBT	$\rho_{\mathrm{Nl,ls}}$	14.4.6.3		728	This is the liquid sodium density at about 1200 K.
1199	RHSLTP	$\rho_{\mathrm{Nl,us}}$	14.4.6.3		0.975 ∗ RHSLBT	
1206	RHSSLQ				6.10^3	This is the density of the steel entrapped in the frozen fuel crust. It is used in LEFREZ to determine the volume of the newly formed crust.
1207	CIBBDI		16.4.3.9.1		0.2	
1208	CIANDI		16.4.3.9.2		0.1	
1210	EGSESO				$8.834 \cdot 10^6$	Used in the functions $T\,(h)$ and $H_{\mathrm{se}}\,(t)$. Should correspond to the steel solidus temperature TME.
1211	EGSELQ				EGSESO+ UEMELT(1	Used in the functions T_{se} left(h right)` and $H_{\mathrm{se}}\,(T)$. Should correspond to the steel liquidus temperature TME + DETEMS.
1212	CPSE	$c_{\mathrm{p,se}}$	16.4.3.10.1.	Eq. (16.4-319)	774.0	
1213	FRMRSE		16.5.3.2	Eq. (16.5-17)	0.5	
1215	RHSSSO				$6.95 \cdot 10^{-3}$	
1216	RGFV	R_{fv}	16.4.3.6	Eq. (16.4-101) Eq. (16.4-118)	31	
1217	RGSV	R_{sv}	16.4.3.7		148	
1229	FNDISR		16.2.3	Eq. (16.2-1)	0.9	
1231	SRFMLE				1.0	This input value maintains the sodium in the structure film when initiating LEVITATE. If SRFMLE = 0 the film on the structure is ignored.

continues on next page

Table 16.8.1 – continued from previous page

Input Location	FORTRAN Variable	Symbol	Section Reference	Equation Reference	Suggested Value (MKS)	Comments
1280	ASRALU		16.4.3.9.3		1	The aspect ratio of solid chunks, defined as $L/2R$.
1284	RALUDI		16.4.3.9.3		0.0	This is the radius of the chunks generated by disruption of the fuel pin. If 0.0, the code will determine the appropriate radius using the local geometry.
1285	RALUFZ		16.4.3.9.3		0.0	This is the radius of the chunks generated by the frozen crust break-up and bulk fuel freezing. If 0.0, the code will determine the appropriate radius using the local geometry.
Block 51, INPCHN						
71	NRPI1				NPIN	All pins are assumed to fail coherently.
72	NRPI2				0	This input constant should be zero whenever NRP11 = NPIN is desired.
182	IMOMEN		16.4.3.8.1 *16.4.3.8.2*	Eq. (16.4-157) Eq. (16.4-193)	0	
Block 65, FUELIN						
2	FMELTM	14.1.2			0.2	
19	FCFAIL				0.0	The molten fuel cavity pressure calculated by DEFORM will be used.

16.9 Output Description

16.9.1 Regular Output

The LEVITATE output has been designed to provide the essential information about the subassembly at a given point in time. The output is printed from LEVDRV and can be obtained at equal time intervals, by specifying the input quantity DTPLP. Additional output can be obtained by specifying the input integers IPGO, IPSTOP and IPNEW. The LEVITATE output will then be printed between cycles IPGO and IPSTOP every IPNEW cycles. It is emphasized that this section describes the output produced by the chunk version. The output produced by the initial release version is very similar, but a number of chunk-related variables are not shown. Some chunk-related variables are shown in the output, but their values remain zero at all times.

The regular output is divided into two large sections, one containing information about the fuel-pin cavity and the other one - significantly more extensive - containing information about all the components present in the coolant channel.

The first line in the pin-related output Figure 16.9.1 contains the computational cycle number and the current time, TIMEPL. This time is measured from the initiation of the out-of-pin fuel motion due to pin failure in the given channel.

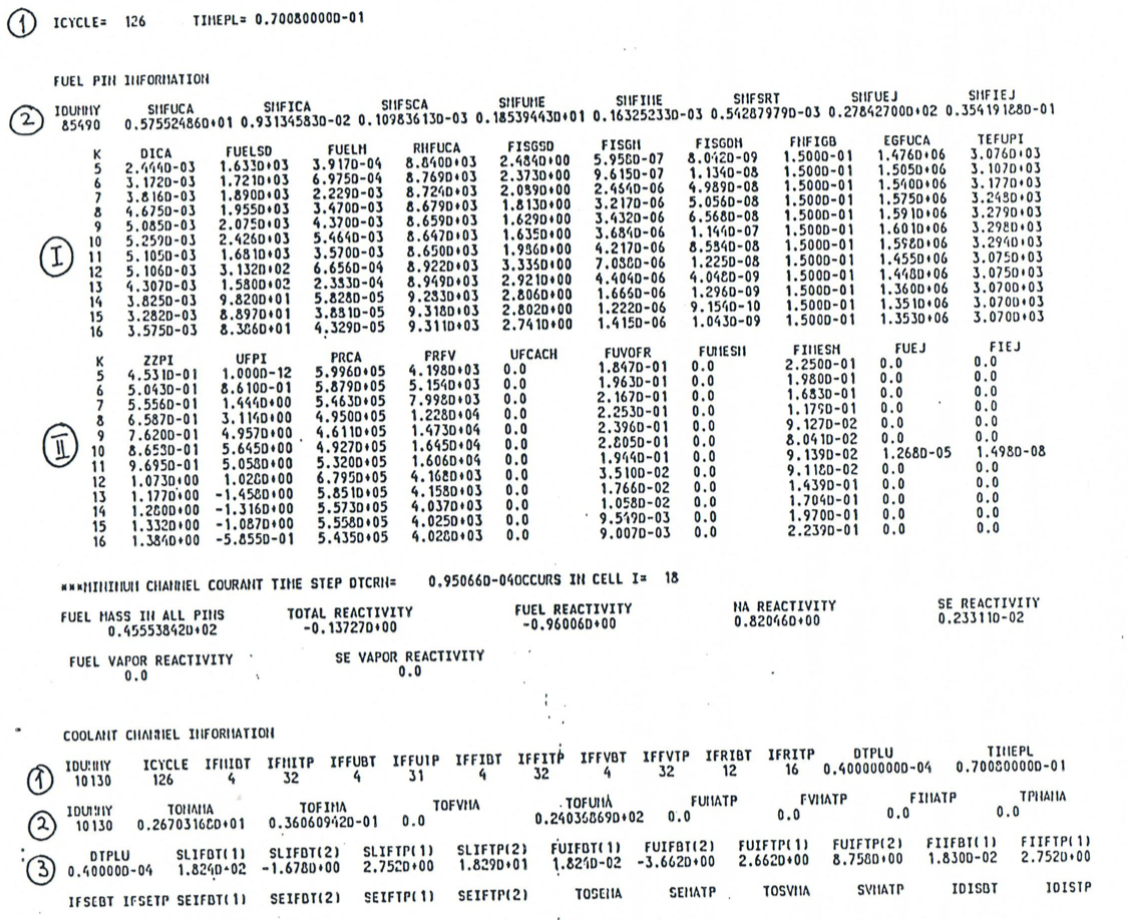
The second line contains some summary information about the fuel-pin cavity, as described below. All these quantities refer to the whole subassembly:

SMFUCA	Total mass of molten fuel in the pin cavity, kg.
SMFICA	Total mass of free fission gas in the pin cavity, kg.
SMFSCA	Total mass of dissolved fission gas in the pin cavity, kg.
SMFUME	Total mass of pin fuel that has molten since LEVIATE initiation, kg.
SMFIME	Total mass of free fission gas released to the cavity due to fuel melting, kg.
SMFSRT	Total mass of fission gas that was originally dissolved in the molten fuel but was released in the meantime, kg.
SMFUEJ	Total amount of molten fuel ejected from the cavity into the coolant channel, kg.
SMFIEJ	Total amount of free fission gas ejected from the cavity into the coolant channel, kg.

Two groups of columns follow, providing more detailed information for all axial cells in the cavity. These columns are described below. Whenever masses are involved, they refer to the whole subassembly, rather than to a single pin.

First group of columns:

k	The number of the axial cell in the cavity. This number refers to the cavity grid, which differs form the channel grid by the integer IDIFF, i.e., k = I - IDIFF.
DICA	Diameter of the cavity, m.
FUELSD	Molten fuel density smeared over the cavity area, kg/m^3.
FUELM	Molten fuel mass, kg.
RHFUCA	Physical density of the molten fuel, kg/m^3.
FISGSD	Free fission-gas density, smeared over the cavity area, kg/m^3.
FISGM	Free fission-gas mass, kg.
FISGDM	Dissolved fission-gas mass, kg.
FNFIGB	Fraction of the fission gas which is released instantaneously upon fuel melting. This fraction is currently an input constant, independent of the axial location.
EGFUCA	Enthalpy of the molten fuel in the cavity, J/kg.
TEFUPI	Temperature of the molten fuel, K



(1) ICYCLE= 126 TIMEPL= 0.700800000-01

FUEL PIN INFORMATION

(2)

IDUMMY	SIGFUCA	SIGFICA	SIGFSCA	SIGFUME	SIGFIME	SIGFSRT	SIGFUEJ	SIGFIEJ
85490	0.575524860+01	0.931345830-02	0.109836130-03	0.185394430+01	0.163252330-03	0.542879790-03	0.278427000+02	0.354191280-01

(I)

K	DICA	FUELSD	FUELM	RHFUCA	FISGSD	FISGM	FISGDM	FMFIGB	EGFUCA	TEFUPI
5	2.444D-03	1.633D+03	3.917D-04	8.840D+03	2.484D+00	5.956D-07	8.042D-09	1.500D-01	1.476D+06	3.076D+03
6	3.172D-03	1.721D+03	6.975D-04	8.769D+03	2.373D+00	9.615D-07	1.134D-08	1.500D-01	1.505D+06	3.107D+03
7	3.816D-03	1.890D+03	2.229D-03	8.724D+03	2.089D+00	2.464D-06	4.989D-08	1.500D-01	1.540D+06	3.177D+03
8	4.675D-03	1.955D+03	3.470D-03	8.679D+03	1.813D+00	3.217D-06	5.056D-08	1.500D-01	1.575D+06	3.248D+03
9	5.085D-03	2.075D+03	4.370D-03	8.659D+03	1.629D+00	3.432D-06	6.568D-08	1.500D-01	1.591D+06	3.279D+03
10	5.259D-03	2.426D+03	5.464D-03	8.647D+03	1.635D+00	3.684D-06	1.144D-07	1.500D-01	1.601D+06	3.298D+03
11	5.105D-03	1.681D+03	3.570D-03	8.650D+03	1.986D+00	4.217D-06	8.584D-08	1.500D-01	1.598D+06	3.294D+03
12	5.106D-03	3.132D+02	6.656D-04	8.922D+03	3.336D+00	7.086D-06	1.225D-08	1.500D-01	1.455D+06	3.075D+03
13	4.307D-03	1.580D+02	2.383D-04	8.949D+03	2.921D+00	4.404D-06	4.048D-09	1.500D-01	1.448D+06	3.075D+03
14	3.825D-03	9.820D+01	5.828D-05	9.233D+03	2.806D+00	1.666D-06	1.296D-09	1.500D-01	1.360D+06	3.070D+03
15	3.282D-03	8.897D+01	3.881D-05	9.318D+03	2.802D+00	1.222D-06	9.154D-10	1.500D-01	1.351D+06	3.070D+03
16	3.575D-03	8.386D+01	4.329D-05	9.311D+03	2.741D+00	1.415D-06	1.043D-09	1.500D-01	1.353D+06	3.070D+03

(II)

K	ZZPI	UFPI	PRCA	PRFV	UFCACH	FUVOFR	FUMESH	FIMESH	FUEJ	FIEJ
5	4.531D-01	1.000D-12	5.996D+05	4.198D+03	0.0	1.847D-01	0.0	2.250D-01	0.0	0.0
6	5.043D-01	8.610D-01	5.879D+05	5.154D+03	0.0	1.963D-01	0.0	1.980D-01	0.0	0.0
7	5.556D-01	1.444D+00	5.463D+05	7.998D+03	0.0	2.167D-01	0.0	1.683D-01	0.0	0.0
8	6.587D-01	3.114D+00	4.950D+05	1.228D+04	0.0	2.253D-01	0.0	1.179D-01	0.0	0.0
9	7.620D-01	4.957D+00	4.611D+05	1.473D+04	0.0	2.396D-01	0.0	9.127D-02	0.0	0.0
10	8.653D-01	5.645D+00	4.927D+05	1.645D+04	0.0	2.805D-01	0.0	8.041D-02	0.0	0.0
11	9.695D-01	5.058D+00	5.320D+05	1.606D+04	0.0	1.944D-01	0.0	9.139D-02	1.268D-05	1.498D-08
12	1.073D+00	1.028D+00	6.795D+05	4.168D+03	0.0	3.510D-02	0.0	9.118D-02	0.0	0.0
13	1.177D+00	-1.458D+00	5.851D+05	4.158D+03	0.0	1.766D-02	0.0	1.439D-01	0.0	0.0
14	1.280D+00	-1.316D+00	5.573D+05	4.037D+03	0.0	1.058D-02	0.0	1.704D-01	0.0	0.0
15	1.332D+00	-1.087D+00	5.558D+05	4.025D+03	0.0	9.549D-03	0.0	1.970D-01	0.0	0.0
16	1.384D+00	-5.855D-01	5.435D+05	4.028D+03	0.0	9.007D-03	0.0	2.239D-01	0.0	0.0

***MINIMUM CHANNEL COURANT TIME STEP DTCRN= 0.95066D-04OCCURS IN CELL I= 18

FUEL MASS IN ALL PINS	TOTAL REACTIVITY	FUEL REACTIVITY	NA REACTIVITY	SE REACTIVITY
0.455538420+02	-0.137270+00	-0.96006D+00	0.820460+00	0.233110-02

FUEL VAPOR REACTIVITY	SE VAPOR REACTIVITY
0.0	0.0

COOLANT CHANNEL INFORMATION

(1)

IDUMMY	ICYCLE	IFMIDT	IFMITP	IFFUBT	IFFUIP	IFFIDT	IFFITP	IFFVBT	IFFVTP	IFRIBT	IFRITP	DTPLU	TIMEPL
10130	126	4	32	4	31	4	32	4	32	12	16	0.400000000-04	0.700800000-01

(2)

IDUMMY	TOMAMA	TOFIMA	TOFVMA	TOFUMA	FUMATP	FVMATP	FIMATP	TPMAMA
10130	0.267031600+01	0.360609420-01	0.0	0.240368690+02	0.0	0.0	0.0	0.0

(3)

DTPLU	SLIFDT(1)	SLIFDT(2)	SLIFTP(1)	SLIFTP(2)	FUIFDT(1)	FUIFDT(2)	FUIFTP(1)	FUIFTP(2)	FIIFDT(1)	FIIFTP(1)
0.400000-04	1.824D+02	-1.678D+00	2.752D+00	1.829D+01	1.824D-02	-3.662D+00	2.662D+00	8.758D+00	1.830D-02	2.752D+00

IFSEDT	IFSETP	SEIFDT(1)	SEIFDT(2)	SEIFTP(1)	SEIFTP(2)	TOSEMA	SEMATP	TOSVMA	SVMATP	IDISDT	IDISTP

Figure 16.9.1: Output Description

The second group of columns:

k	The number of the axial cell in the cavity.
ZZPI	Axial location of the lower boundary of cell k, measured from the bottom of the pin, m.
UFPI	Velocity of the molten fuel/fission gas mixture in the cavity, at the axial location ZZPI, m/s.
PRCA	Total pressure in the cavity, Pa.
PRFV	Fuel vapor pressure in the cavity, Pa.
UFCACH	Radial velocity of fuel/gas mixture being ejected from the pin cavity into the channel, m/s. This velocity is only calculated when the mechanistic ejection model is used, i.e., INRAEJ=1.
FUVOFR	Volume fraction of molten fuel in the pin cavity.
FUMESM	Mass of fuel molten during the current time step, kg.
FIMESM	Mass of free fission gas added to cell K of cavity during the current time step, kg.
FUEJ	Mass of fuel ejected from cavity during the current time step, kg.
FIEJ	Mass of free fission gas ejected from cavity during the current time step, kg.

An important characteristic LEVITATE is that the fuel pins can be disrupted at certain axial locations. The disrupted nodes can be identified by zeros in the columns headed: FUELSD, FUELM, FISGSD, FISGM, FISGDM, FUMESM, FIMESM, FUEJ and FIEJ. The other columns are not zeroed out, but the numbers appearing in the locations corresponding to disrupted nodes should be disregarded. One exception is the column PRCA (cavity pressure). In this column the nodes adjacent to the still intact pin stubs always contain a pressure. This is the pressure in the channel and was used in the in-pin momentum calculation as a boundary condition. The column headed PRVI (viscous pressure) should be totally disregarded, as the viscous pressure is not used in LEVITATE. Finally, the columns FUEJ and FIEJ (fuel and fission gas ejected during the last cycle) can occasionally contain negative quantities in the LEVITATE output. This is indicative of fuel and fission gas reentry in the pin cavity, via the open ends of the still intact pin stubs. The quantities SMFUEJ and SMEIEJ (total amount of fuel and fission gas ejected from the pin) have *not* yet been modified to account for the axial ejection via the pin stubs. Thus, after pin disruption, the amount of fuel ejected SMFUEJ might be different from the total amount of fuel in the channel, TOFUMA (to be described below).

The printout about the fuel pins is followed by two lines of summary information, listing the fuel mass in all the pins in the subassembly and the various reactivity components for the channel being printed, in dollars.

The section on the coolant channel follows. The first 6 lines contain information about the boundaries of various component regions and several integral quantities (Figure 16.9.1 and Figure 16.9.2). This summary information is followed by detailed information about each cell in the LEVITATE region. This information is printed in columns which are described below.

Summary information (left to right and top to bottom):

ICYCLE	The cycle number. Note that it starts from 0, because the increment is performed after the main printout.
IFMIBT, IFMITP	Bottom and top cells of the interaction region where LEVITATE performs calculations. This region is bounded by the liquid sodium slugs.
IFFUBT, IFFUTP	Bottom and top cell of the molten fuel region.
IFFIBT, IFFITP	Bottom and top cell of the fission gas region.
IFFVBT, IFFVTP	Bottom and top cell of the fission fuel vapor region.
IFRIBT, IFRITP	Bottom and top cell of the fuel-pin rip.
DTPLU	Time step used in this cycle.

continues on next page

Table 16.9.1 – continued from previous page

TIMEPL	Current LEVITAT time, measured since the fuel motion initiation in this channel.
TONAMA	Total sodium mass in the channel.
TOFIMA	Total fission-gas mass in the channel.
TOFVMA	Total fuel vapor mass in the channel.
TOFUMA	Total molten fuel mass in the channel.
FUMATP	Molten fuel mass in the upper plenum.
FVMATP	Fuel vapor mass in the upper plenum.
FIMATP	Fission-gas mass in the upper plenum.
TPNAMA	Sodium mass in the upper plenum.
SLIFBT (1), SLIFBT (2)	Location and velocity of the upper boundary of the lower sodium slug.
SLIFTP (1), SLIFTP (2)	Location and velocity of the lower boundary of the upper sodium slug.
FUIFBT (1), FUIFTP (2)	Location and velocity of the lower boundary of the molten fuel region.
FUIFBT (1), FUIFTP (2)	Same as above, for the upper boundary.
FIFFBT (1), FIIFTP (2)	Location of the lower and upper interface of the fission-gas region.
IFSEBT, IFSETP	Bottom and top cell of the molten steel region.
SEIFBT (1), SEIFBT (2)	Location and velocity of the lower boundary of the molten steel region.
SEIFTP (1), SEIFTP (2)	Same as above, for the upper boundary.
TOSEMA	Total mass of molten steel in the channel.
SEMATP	Molten steel mass in the upper plenum.
TOSVMA	Total mass of steel vapor in the channel.
SVMATP	Steel vapor mass in the upper plenum.
IDISBT, IDISTP	Bottom and top cell of the disrupted pin region.
IFLUBT, IFLUTP	Bottom and top cell of the chunk region.
FLFBT (1), FLFBT (2)	Location and velocity of the lower boundary of the solid chunk region.
FLIFTP (1), FLIFTP (2)	Same as above for the upper boundary.
TOLUMA	Total mass of fuel chunks in the channel.
FLMATP	Fuel chunk mass in the upper plenum.
PRIN	Inlet pressure (at the bottom of the lower slug).
PREX	Outlet pressure (at the top of the upper slug).
IFSVBT, IFSVTP	Bottom and top cell of the steel vapor region.
SVIFBT (1), SVIFTP (2)	Location and velocity of the lower boundary of the steel vapor region.
SVIFTP (1), SVIFTP (2)	Same as above, for the upper boundary.
FVIFBT (1), FVIFTP (1)	Location of the lower and upper boundaries of the fuel vapor region.
TOSLMA	Total mass of steel chunks in the channel.
SLMATP	Steel chunk mass in the upper plenum.

The first group of columns (Figure 16.9.2):

I	The current cell number.
ZC	Axial location of the lower boundary of cell i, measured from the bottom of the pin.
UMCH	Velocity of the gas mixture at boundary I, in cell I. Note that because of the use of dual velocities at the cell boundaries, it is necessary to specify whether the velocity printed for boundary I is in cell I or in cell $I-1$.
UFCH	Velocity of the molten fuel *and* molten steel at boundary I, in cell I.
PRCH	Total pressure in cell I.
SDNA	Density of sodium smeared over the open channel.
FISD	Density of fission gas smeared over the open channel.
THSECH	Volume fraction occupied by molten steel in cell I. This fraction always refers to the reference channel, with area AXMX.
THCHOP	Volume fraction occupied by the open channel. It is suggested that AXMX be selected such that the original THCHOP $(I) = 1$.
THFUCH	Volume fraction occupied by the molten fuel in cell i.
THNAFM	Volume fraction of the sodium film. Not used in LEVITATE currently; should always be zero.
THNL	Volume fraction of liquid sodium.

The second group of columns (Figure 16.9.2 and Figure 16.9.3).

I	The current cell number.
IFLAG	Flow regime indicator. Has the following significance: 2 - annular steel regime; 3 - annular fuel regime; 4 - bubbly fuel region; 5 - bubbly steel regime.
ZC	Same as previously described.
TEFUOS	Temperature of the molten fuel in cell I.
TENA	Temperature of the sodium vapor and fission gas.
TECLOS	Temperature of the outer cladding node, K.
TECLIN	Temperature of inner cladding node. If this quantity is - 100, it indicates that the inner node has been removed due to ablation, K.
TESROS	Temperature of the “outer” structure node, i.e., the node facing the channel, K.
TESRIN	Temperature of the “inner” structure node, K.
PRNV	Saturation pressure of sodium at TENA. It is meaningful only for the nodes where two-phase sodium is present, i.e., THNL$(I) \neq O$.
FUCH	Fuel mass in cell I, in kg.
CFFFCL	Fraction of cladding perimeter (or area) covered by the frozen fuel crust, in cell i.

The third group of columns (Figure 16.9.3):

I	The current cell number.
IFLAG	Described above.
ZC	Described above.
DESECH	Generalized steel density: $\rho'_{\text{se,i}} = \rho_{\text{se}} \frac{A_{\text{se,i}}}{\text{AXMX}}$.
TESECH	Temperature of the molten steel, K.
DEFUCL	Generalized density of frozen fuel on the clad.
DEFUSR	Generalized density of frozen fuel on the hexcan wall.
DESECL	Generalized density of steel entrapped in the fuel crust on the clad.
DESESR	Generalized density of steel entrapped in the fuel crust on the structure.
CFFFSR	Fraction of structure perimeter (or area) covered by frozen fuel.
TEFFCL	Temperature of the frozen fuel crust on the cladding.
TEFFSR	Temperature of the frozen fuel crust on the structure.

The fourth group of columns (Figure 16.9.3 and Figure 16.9.4):

I	The current cell number.
IDISR	Pin disruption indicator. Has the following significance: 0 - no disruption and no pin rip present; 1 - total disruption of the pins has occurred at this location; 2 - pin undisrupted but ripped. No injection taking place; 3 - pin undisrupted but ripped. Injection of molten fuel in the channel has occurred during this cycle.
ZC	Described above.
WICLAD	Thickness of the cladding in cell I. If zero, the pin has been disrupted (IDISR $= 1$) or the cladding was totally ablated (IDISR $\neq 1$, TECLIN $= -100$).
WISTRC	Thickness of the structure in cell I.
TEFFSS	Surface temperature of the frozen fuel crust on the structure.
DELUCH	Generalized density of fuel chunks, kg/m^3.
TELUCH	Temperature of fuel chunks, K.
ULCH	Velocity of fuel chunks at boundary I (no dual velocities are used for the chunks).
THLUCH	Volume fraction occupied by the fuel and steel chunks.
RALUCH	Radius of chunks in cell I, m.
XNLUCH	Number of chunks in cell I.

The fifth group of columns (Figure 16.9.4 and Figure 16.9.5):

I	The current cell number.
IDISR	Described above.
ZC	Described above.
DEFVCH	Generalized density of the fuel vapor, kg/m^3.
DESVCH	Generalized density of the steel vapor, kg/m^3.
TEFUVA	Temperature of fuel vapor, K.
PRFV	Partial pressure of fuel vapor, Pa.
PRSV	Partial pressure of steel vapor, Pa.
TESEVA	Temperature of steel vapor, K.
PRFI	Partial pressure of fission gas, Pa.
PRNA	Partial pressure of the sodium vapor, Pa.
RHFULU	Density of fuel chunks. It can be different from the theoretical fuel density due to porosity.

The sixth group of columns: (This group refers entirely to the fuel/steel chunks and was not present in the initial release version).

I	The current cell number.
IDISR	Described above.
ZC	Described above.
DESELU	Generalized density of steel chunks, kg/m^3.
TESELU	Temperature of steel chunks, K.
FRSELU	Volume fraction of steel in the fuel/steel chunks. Can vary from 0 to 1.
DEFILU	Generalized density of fission gas associated with the fuel chunks.

The seventh group of columns has the title "Temperature Map of Region Outside the Interaction Zone." This output provides information about temperatures in the liquid sodium slug regions. It was not available in the Release 1.0 version of LEVITATE. It should be noted that these values are calculated at the end of the last heat-transfer time which can be fractions of a millisecond before the time of the current printout.

I	Axial channel index.
ZCOOL	Location of the lower boundary of mesh cell I.
TREFL2(2)	Inner reflector node temperature. Reflectors can be located only below or above the pin zone K2PIN.
TREFL2(1)	Temperature of the outer reflector node which is facing the coolant.
T1(NEPP)	Inner cladding surface temperature (i.e., next to the fuel).
T1(NE)	Cladding temperature of the middle cladding node.
T1(NEP)	Outer cladding surface temperature (i.e., next to the coolant).
TENA	Sodium temperature.
TSAT	Sodium saturation temperature. This is calculated only for the pin zone to detect sodium-boiling initiation.
TSTR2(1)	Temperature of the structure node facing the coolant channel.
TSTR2(2)	Temperature of the structure node facing the neighboring hexcan wall.
PRCH	Pressure in the coolant channel. This is calculated outside the fuel-pin zone only if the interaction region extends beyond it.

16.9.2 Auxiliary and Debug Output

Several auxiliary WRITE statement are incorporated in LEVITATE. These statements are activated only when some special situation occurs. The definition of the term "special situation" is somewhat loose, and it is possible that the situation indicated by the WRITE statement printout is legitimate for the given run. The user should attempt to familiarize himself with the meaning of these messages. Their general form is:

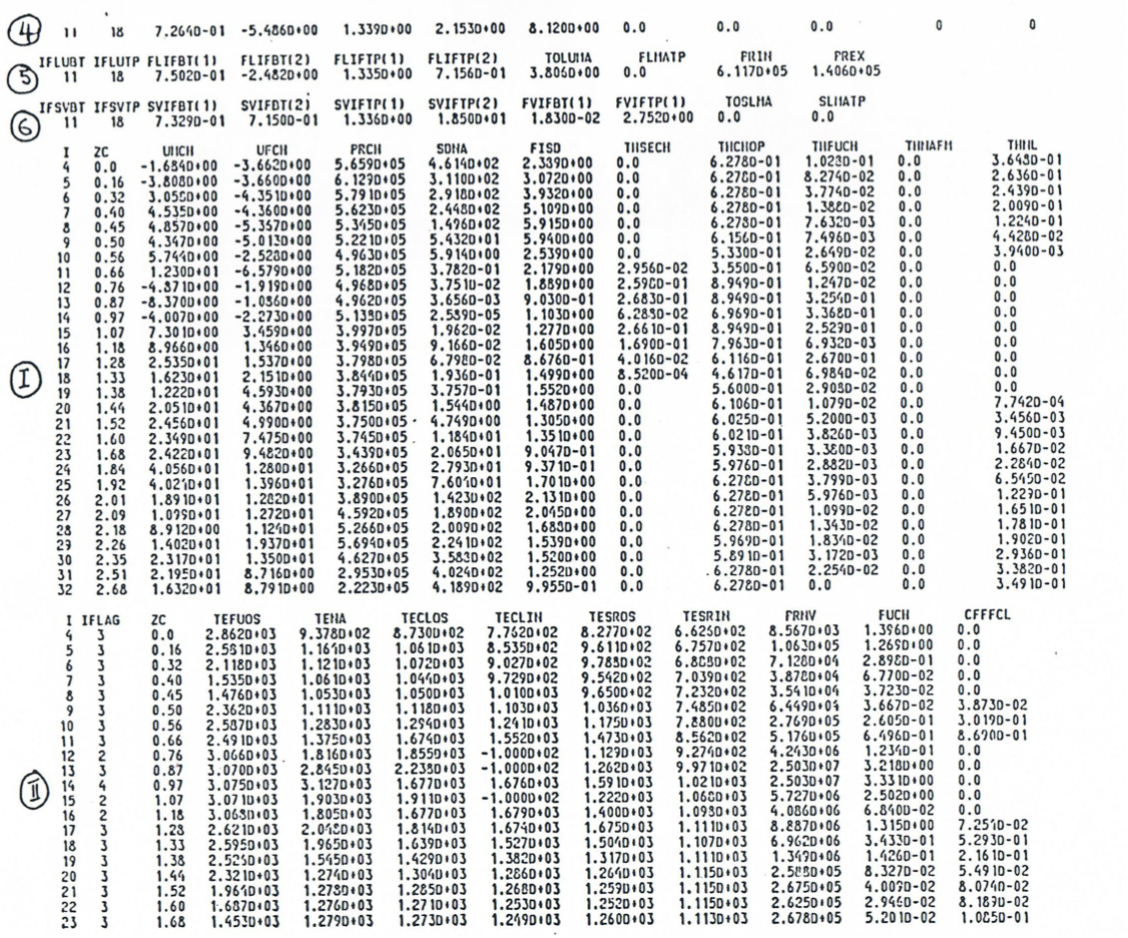
*** Brief explanatory message. XXXX-999***

where XXXX are the first four letters in the name of the routine printing the message and 999 is the label of the FORMAT statement printing the message. Several integers and floating point numbers may follow. They depend on the list of variables associated with the particular WRITE statement. An example of such messages which are used for information only, rather than to indicate a problem, are the messages

*** LEVITATE STARTS ***

and

*** LEVITATE ENDS ***



④

11	18	7.2640-01	-5.4860+00	1.3390+00	2.1530+00	8.1200+00	0.0	0.0	0.0	0	0

⑤

IFLUBT	IFLUTP	FLIFBT(1)	FLIFBT(2)	FLIFTP(1)	FLIFTP(2)	TOLUNA	FLMATP	FRIN	FREX
11	18	7.5020-01	-2.4820+00	1.3350+00	7.1560-01	3.8060+00	0.0	6.1170+05	1.4060+05

⑥

IFSVBT	IFSVTP	SVIFBT(1)	SVIFBT(2)	SVIFTP(1)	SVIFTP(2)	FVIFBT(1)	FVIFTP(1)	TOSLHA	SLHATP
11	18	7.3290-01	7.1500-01	1.3360+00	1.8500+01	1.8300-02	2.7520+00	0.0	0.0

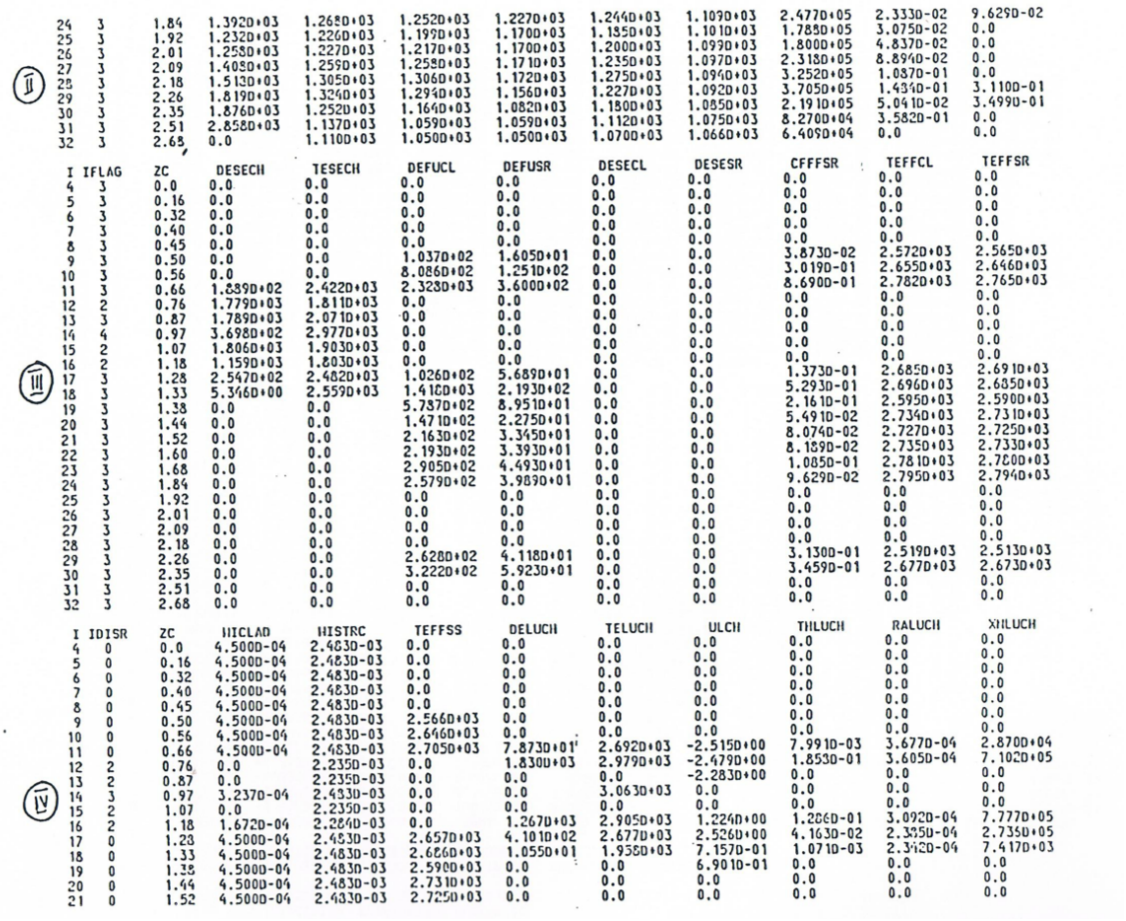
Ⓘ

I	ZC	UNCH	UFCH	PRCH	SDNA	FISD	THSECH	THCHOP	THFUCH	THNAFH	THNL
4	0.0	-1.6840+00	-3.6620+00	5.6590+05	4.6140+02	2.3390+00	0.0	6.2780-01	1.0230-01	0.0	3.6430-01
5	0.16	-3.8080+00	-3.6600+00	6.1290+05	3.1100+02	3.0720+00	0.0	6.2760-01	8.2740-02	0.0	2.6360-01
6	0.32	3.0560+00	-4.3510+00	5.7910+05	2.9180+02	3.9320+00	0.0	6.2780-01	3.7740-02	0.0	2.4390-01
7	0.40	4.5350+00	-4.3600+00	5.6230+05	2.4480+02	5.1090+00	0.0	6.2780-01	1.3820-02	0.0	2.0090-01
8	0.45	4.8570+00	-5.3570+00	5.3450+05	1.4960+02	5.9150+00	0.0	6.2780-01	7.6320-03	0.0	1.2240-01
9	0.50	4.3470+00	-5.0130+00	5.2210+05	5.4320+01	5.9400+00	0.0	6.1560-01	7.4960-03	0.0	4.4280-02
10	0.56	5.7440+00	-2.5280+00	4.9630+05	5.9140+00	2.5390+00	0.0	5.3300-01	2.6490-02	0.0	3.9400-03
11	0.66	1.2300+01	-6.5790+00	5.1820+05	3.7820-01	2.1790+00	2.9560-02	3.5500-01	6.5900-02	0.0	0.0
12	0.76	-4.8710+00	-1.9190+00	4.9680+05	3.7510-02	1.8890+00	2.5900-01	8.9490-01	1.2470-02	0.0	0.0
13	0.87	-8.3700+00	-1.0360+00	4.9620+05	3.6560-03	9.0300-01	2.6830-01	8.9490-01	3.2540-01	0.0	0.0
14	0.97	-4.0070+00	-2.2730+00	5.1350+05	2.5390-05	1.1030+00	6.2850-02	6.9690-01	3.3680-01	0.0	0.0
15	1.07	7.3010+00	3.4590+00	3.9970+05	1.9620-02	1.2770+00	2.6610-01	8.9490-01	2.5290-01	0.0	0.0
16	1.18	8.9660+00	1.3460+00	3.9490+05	9.1660-02	1.6050+00	1.6900-01	7.9630-01	6.9320-03	0.0	0.0
17	1.28	2.5350+01	1.5370+00	3.7980+05	6.7980-02	8.6760-01	4.0160-02	6.1160-01	2.6780-01	0.0	0.0
18	1.33	1.6230+01	2.1510+00	3.8440+05	1.9360-01	1.4990+00	8.5200-04	4.6170-01	6.9840-02	0.0	0.0
19	1.38	1.2220+01	4.5930+00	3.7930+05	3.7570-01	1.5520+00	0.0	5.6000-01	2.9050-02	0.0	0.0
20	1.44	2.0510+01	4.3670+00	3.8150+05	1.5440+00	1.4870+00	0.0	6.1060-01	1.0790-02	0.0	7.7420-04
21	1.52	2.4560+01	4.9900+00	3.7500+05	4.7490+00	1.3050+00	0.0	6.0250-01	5.2000-03	0.0	3.4560-03
22	1.60	2.3490+01	7.4750+00	3.7450+05	1.1840+01	1.3510+00	0.0	6.0210-01	3.8260-03	0.0	9.4500-03
23	1.68	2.4220+01	9.4820+00	3.4390+05	2.0650+01	9.0470-01	0.0	5.9330-01	3.3800-03	0.0	1.6670-02
24	1.84	4.0560+01	1.2800+01	3.2660+05	2.7930+01	9.3710-01	0.0	5.9760-01	2.8820-03	0.0	2.2840-02
25	1.92	4.0210+01	1.3960+01	3.2760+05	7.6040+01	1.7010+00	0.0	6.2780-01	3.7990-03	0.0	6.5450-02
26	2.01	1.8910+01	1.2820+01	3.8900+05	1.4230+02	2.1310+00	0.0	6.2780-01	5.9760-03	0.0	1.2290-01
27	2.09	1.0990+01	1.2720+01	4.5920+05	1.8900+02	2.0450+00	0.0	6.2780-01	1.0990-02	0.0	1.6510-01
28	2.18	8.9120+00	1.1240+01	5.2660+05	2.0090+02	1.6830+00	0.0	6.2780-01	1.3430-02	0.0	1.7810-01
29	2.26	1.4020+01	1.9370+01	5.6940+05	2.2410+02	1.5390+00	0.0	5.9690-01	1.8340-02	0.0	1.9020-01
30	2.35	2.3170+01	1.3500+01	4.6270+05	3.5830+02	1.5200+00	0.0	5.8910-01	3.1720-03	0.0	2.9360-01
31	2.51	2.1950+01	8.7160+00	2.9530+05	4.0240+02	1.2520+00	0.0	6.2780-01	2.2540-02	0.0	3.3820-01
32	2.68	1.6320+01	8.7910+00	2.2230+05	4.1890+02	9.9550-01	0.0	6.2780-01	0.0	0.0	3.4910-01

Ⓘ

I	IFLAG	ZC	TEFUOS	TENA	TECLOS	TECLIN	TESROS	TESRIN	FRHV	FUCH	CFFFCL
4	3	0.0	2.8620+03	9.3780+02	8.7300+02	7.7620+02	8.2770+02	6.6250+02	8.5670+03	1.3960+00	0.0
5	3	0.16	2.5810+03	1.1610+03	1.0610+03	8.5350+02	9.6110+02	6.7570+02	1.0630+05	1.2690+00	0.0
6	3	0.32	2.1180+03	1.1210+03	1.0720+03	9.0270+02	9.7850+02	6.8080+02	7.1280+04	2.8960-01	0.0
7	3	0.40	1.5350+03	1.0610+03	1.0440+03	9.7290+02	9.5420+02	7.0390+02	3.8780+04	6.7700-02	0.0
8	3	0.45	1.4760+03	1.0530+03	1.0500+03	1.0100+03	9.6500+02	7.2320+02	3.5410+04	3.7230-02	0.0
9	3	0.50	2.3620+03	1.1110+03	1.1180+03	1.1030+03	1.0360+03	7.4850+02	6.4490+04	3.6670-02	3.8730-02
10	3	0.56	2.5870+03	1.2830+03	1.2940+03	1.2410+03	1.1750+03	7.8800+02	2.7690+05	2.6050-01	3.0190-01
11	3	0.66	2.4910+03	1.3750+03	1.6740+03	1.5520+03	1.4730+03	8.5620+02	5.1760+05	6.4960-01	8.6900-01
12	2	0.76	3.0660+03	1.8160+03	1.8559+03	-1.0000+02	1.1290+03	9.2740+02	4.2430+06	1.2340-01	0.0
13	3	0.87	3.0700+03	2.8450+03	2.2380+03	-1.0000+02	1.2620+03	9.9710+02	2.5030+07	3.2180+00	0.0
14	4	0.97	3.0750+03	3.1270+03	1.6770+03	1.6760+03	1.5910+03	1.0210+03	2.5030+07	3.3310+00	0.0
15	2	1.07	3.0710+03	1.9030+03	1.9110+03	-1.0000+02	1.2220+03	1.0650+03	5.7270+06	2.5020+00	0.0
16	2	1.18	3.0630+03	1.8050+03	1.6770+03	1.6790+03	1.4000+03	1.0930+03	4.0860+06	6.8400-02	0.0
17	3	1.28	2.6210+03	2.0420+03	1.8140+03	1.6740+03	1.6750+03	1.1110+03	8.8870+06	1.3150+00	7.2540-02
18	3	1.33	2.5950+03	1.9650+03	1.6390+03	1.5270+03	1.5040+03	1.1070+03	6.9620+06	3.4330-01	5.2930-01
19	3	1.38	2.5290+03	1.5450+03	1.4290+03	1.3820+03	1.3170+03	1.1110+03	1.3490+06	1.4260-01	2.1610-01
20	3	1.44	2.3210+03	1.2740+03	1.3040+03	1.2860+03	1.2640+03	1.1150+03	2.5880+05	8.3270-02	5.4910-02
21	3	1.52	1.9640+03	1.2730+03	1.2850+03	1.2680+03	1.2590+03	1.1150+03	2.6750+05	4.0090-02	8.0740-02
22	3	1.60	1.6870+03	1.2760+03	1.2710+03	1.2530+03	1.2520+03	1.1150+03	2.6250+05	2.9460-02	8.1830-02
23	3	1.68	1.4530+03	1.2790+03	1.2730+03	1.2490+03	1.2600+03	1.1130+03	2.6780+05	5.2010-02	1.0850-01

Figure 16.9.2: Output Description (First Continuation)



II	24	3	1.84	1.392D+03	1.268D+03	1.252D+03	1.227D+03	1.244D+03	1.109D+03	2.477D+05	2.333D-02	9.629D-02
	25	3	1.92	1.232D+03	1.226D+03	1.199D+03	1.170D+03	1.185D+03	1.101D+03	1.788D+05	3.075D-02	0.0
	26	3	2.01	1.258D+03	1.227D+03	1.217D+03	1.170D+03	1.200D+03	1.099D+03	1.800D+05	4.837D-02	0.0
	27	3	2.09	1.403D+03	1.259D+03	1.258D+03	1.171D+03	1.235D+03	1.097D+03	2.318D+05	8.894D-02	0.0
	28	3	2.18	1.513D+03	1.305D+03	1.306D+03	1.172D+03	1.275D+03	1.094D+03	3.252D+05	1.087D-01	0.0
	29	3	2.26	1.819D+03	1.324D+03	1.294D+03	1.156D+03	1.227D+03	1.092D+03	3.705D+05	1.434D-01	3.110D-01
	30	3	2.35	1.876D+03	1.252D+03	1.164D+03	1.082D+03	1.180D+03	1.085D+03	2.191D+05	5.041D-02	3.499D-01
	31	3	2.51	2.858D+03	1.137D+03	1.059D+03	1.059D+03	1.112D+03	1.075D+03	8.270D+04	3.582D-01	0.0
	32	3	2.68	0.0	1.110D+03	1.050D+03	1.050D+03	1.070D+03	1.066D+03	6.409D+04	0.0	0.0

	I	IFLAG	ZC	DESECH	TESECH	DEFUCL	DEFUSR	DESECL	DESESR	CFFFSR	TEFFCL	TEFFSR
III	4	3	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
	5	3	0.16	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
	6	3	0.32	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
	7	3	0.40	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
	8	3	0.45	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
	9	3	0.50	0.0	0.0	1.037D+02	1.605D+01	0.0	0.0	3.873D-02	2.572D+03	2.565D+03
	10	3	0.56	0.0	0.0	8.086D+02	1.251D+02	0.0	0.0	3.019D-01	2.655D+03	2.646D+03
	11	3	0.66	1.689D+02	2.422D+03	2.323D+03	3.600D+02	0.0	0.0	8.690D-01	2.782D+03	2.765D+03
	12	2	0.76	1.779D+03	1.811D+03	0.0	0.0	0.0	0.0	0.0	0.0	0.0
	13	3	0.87	1.789D+03	2.071D+03	0.0	0.0	0.0	0.0	0.0	0.0	0.0
	14	4	0.97	3.698D+02	2.977D+03	0.0	0.0	0.0	0.0	0.0	0.0	0.0
	15	2	1.07	1.806D+03	1.903D+03	0.0	0.0	0.0	0.0	0.0	0.0	0.0
	16	2	1.18	1.159D+03	1.803D+03	0.0	0.0	0.0	0.0	1.373D-01	2.685D+03	2.691D+03
	17	3	1.28	2.547D+02	2.482D+03	1.026D+02	5.689D+01	0.0	0.0	5.293D-01	2.696D+03	2.685D+03
	18	3	1.33	5.346D+00	2.559D+03	1.416D+03	2.193D+02	0.0	0.0	2.161D-01	2.595D+03	2.590D+03
	19	3	1.38	0.0	0.0	5.787D+02	8.951D+01	0.0	0.0	5.491D-02	2.734D+03	2.731D+03
	20	3	1.44	0.0	0.0	1.471D+02	2.275D+01	0.0	0.0	8.074D-02	2.727D+03	2.725D+03
	21	3	1.52	0.0	0.0	2.163D+02	3.345D+01	0.0	0.0	8.189D-02	2.735D+03	2.733D+03
	22	3	1.60	0.0	0.0	2.193D+02	3.393D+01	0.0	0.0	1.085D-01	2.781D+03	2.780D+03
	23	3	1.68	0.0	0.0	2.905D+02	4.493D+01	0.0	0.0	9.629D-02	2.795D+03	2.794D+03
	24	3	1.84	0.0	0.0	2.579D+02	3.989D+01	0.0	0.0	0.0	0.0	0.0
	25	3	1.92	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
	26	3	2.01	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
	27	3	2.09	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
	28	3	2.18	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
	29	3	2.26	0.0	0.0	2.628D+02	4.118D+01	0.0	0.0	3.130D-01	2.519D+03	2.513D+03
	30	3	2.35	0.0	0.0	3.222D+02	5.923D+01	0.0	0.0	3.459D-01	2.677D+03	2.673D+03
	31	3	2.51	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
	32	3	2.68	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0

	I	IDISR	ZC	HICLAD	HISTRC	TEFFSS	DELUCH	TELUCH	ULCH	THLUCH	RALUCH	XHLUCH
IV	4	0	0.0	4.500D-04	2.483D-03	0.0	0.0	0.0	0.0	0.0	0.0	0.0
	5	0	0.16	4.500D-04	2.483D-03	0.0	0.0	0.0	0.0	0.0	0.0	0.0
	6	0	0.32	4.500D-04	2.483D-03	0.0	0.0	0.0	0.0	0.0	0.0	0.0
	7	0	0.40	4.500D-04	2.483D-03	0.0	0.0	0.0	0.0	0.0	0.0	0.0
	8	0	0.45	4.500D-04	2.483D-03	0.0	0.0	0.0	0.0	0.0	0.0	0.0
	9	0	0.50	4.500D-04	2.483D-03	2.566D+03	0.0	0.0	0.0	0.0	0.0	0.0
	10	0	0.56	4.500D-04	2.483D-03	2.646D+03	0.0	0.0	0.0	0.0	0.0	0.0
	11	0	0.66	4.500D-04	2.483D-03	2.705D+03	7.873D+01	2.692D+03	-2.515D+00	7.991D-03	3.677D-04	2.870D+04
	12	2	0.76	0.0	2.235D-03	0.0	1.830D+03	2.979D+03	-2.479D+00	1.853D-01	3.605D-04	7.102D+05
	13	2	0.87	0.0	2.235D-03	0.0	0.0	0.0	-2.283D+00	0.0	0.0	0.0
	14	3	0.97	3.237D-04	2.483D-03	0.0	0.0	3.063D+03	0.0	0.0	0.0	0.0
	15	2	1.07	0.0	2.235D-03	0.0	0.0	0.0	0.0	0.0	0.0	0.0
	16	2	1.18	1.672D-04	2.284D-03	0.0	1.267D+03	2.905D+03	1.224D+00	1.206D-01	3.092D-04	7.777D+05
	17	0	1.28	4.500D-04	2.483D-03	2.657D+03	4.101D+02	2.677D+03	2.526D+00	4.163D-02	2.335D-04	2.735D+05
	18	0	1.33	4.500D-04	2.483D-03	2.686D+03	1.055D+01	1.958D+03	7.157D-01	1.071D-03	2.342D-04	7.417D+03
	19	0	1.38	4.500D-04	2.483D-03	2.598D+03	0.0	0.0	6.901D-01	0.0	0.0	0.0
	20	0	1.44	4.500D-04	2.483D-03	2.731D+03	0.0	0.0	0.0	0.0	0.0	0.0
	21	0	1.52	4.500D-04	2.483D-03	2.725D+03	0.0	0.0	0.0	0.0	0.0	0.0

Figure 16.9.3: Output Description (Second Continuation)

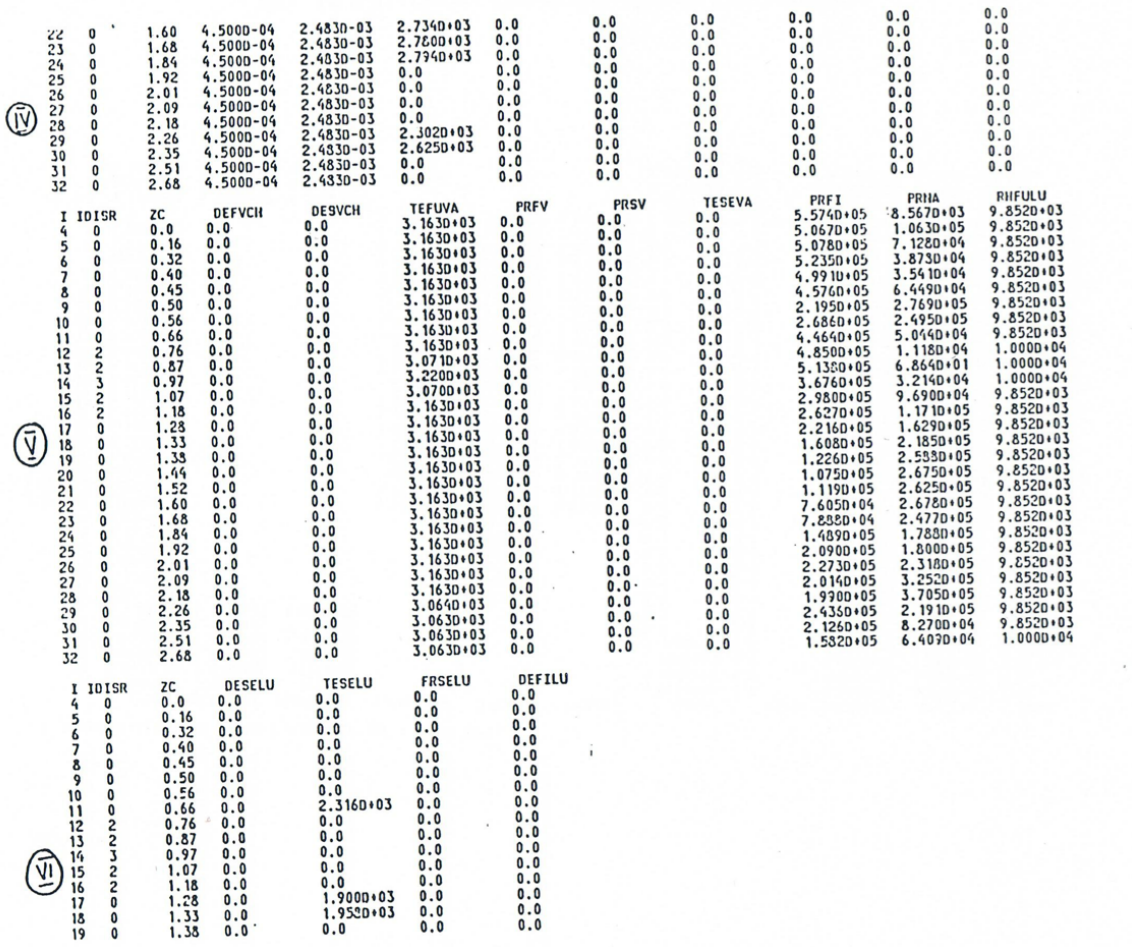
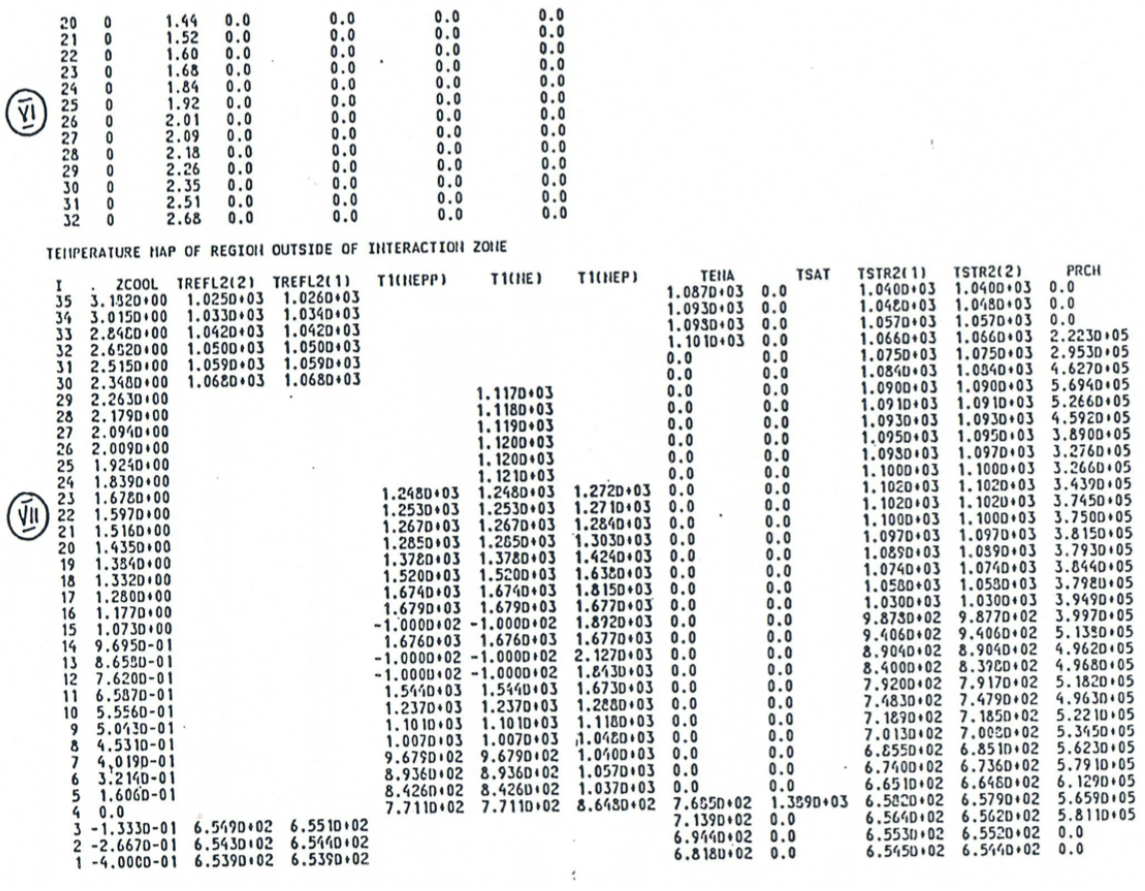


Figure 16.9.4: Output Description (Third Continuation)



20	0	1.44	0.0	0.0	0.0	0.0
21	0	1.52	0.0	0.0	0.0	0.0
22	0	1.60	0.0	0.0	0.0	0.0
23	0	1.68	0.0	0.0	0.0	0.0
24	0	1.84	0.0	0.0	0.0	0.0
25	0	1.92	0.0	0.0	0.0	0.0
26	0	2.01	0.0	0.0	0.0	0.0
27	0	2.09	0.0	0.0	0.0	0.0
28	0	2.18	0.0	0.0	0.0	0.0
29	0	2.26	0.0	0.0	0.0	0.0
30	0	2.35	0.0	0.0	0.0	0.0
31	0	2.51	0.0	0.0	0.0	0.0
32	0	2.68	0.0	0.0	0.0	0.0

TEMPERATURE MAP OF REGION OUTSIDE OF INTERACTION ZONE

I	. ZCOOL	TREFL2(2)	TREFL2(1)	T1(NEPP)	T1(NE)	T1(NEP)	TEMA	TSAT	TSTR2(1)	TSTR2(2)	PRCH
35	3.1820+00	1.0250+03	1.0260+03				1.0870+03	0.0	1.0400+03	1.0400+03	0.0
34	3.0150+00	1.0330+03	1.0340+03				1.0930+03	0.0	1.0480+03	1.0480+03	0.0
33	2.8460+00	1.0420+03	1.0420+03				1.0930+03	0.0	1.0570+03	1.0570+03	0.0
32	2.6520+00	1.0500+03	1.0500+03				1.1010+03	0.0	1.0660+03	1.0660+03	2.2230+05
31	2.5150+00	1.0590+03	1.0590+03				0.0	0.0	1.0750+03	1.0750+03	2.9530+05
30	2.3480+00	1.0680+03	1.0680+03				0.0	0.0	1.0840+03	1.0840+03	4.6270+05
29	2.2630+00				1.1170+03		0.0	0.0	1.0900+03	1.0900+03	5.6940+05
28	2.1790+00				1.1180+03		0.0	0.0	1.0910+03	1.0910+03	5.2660+05
27	2.0940+00				1.1190+03		0.0	0.0	1.0930+03	1.0930+03	4.5920+05
26	2.0090+00				1.1200+03		0.0	0.0	1.0950+03	1.0950+03	3.8900+05
25	1.9240+00				1.1200+03		0.0	0.0	1.0930+03	1.0970+03	3.2760+05
24	1.8390+00				1.1210+03		0.0	0.0	1.1000+03	1.1000+03	3.2660+05
23	1.6780+00			1.2480+03	1.2480+03	1.2720+03	0.0	0.0	1.1020+03	1.1020+03	3.4390+05
22	1.5970+00			1.2530+03	1.2530+03	1.2710+03	0.0	0.0	1.1020+03	1.1020+03	3.7450+05
21	1.5160+00			1.2670+03	1.2670+03	1.2840+03	0.0	0.0	1.1000+03	1.1000+03	3.7500+05
20	1.4350+00			1.2850+03	1.2850+03	1.3030+03	0.0	0.0	1.0970+03	1.0970+03	3.8150+05
19	1.3840+00			1.3780+03	1.3780+03	1.4240+03	0.0	0.0	1.0890+03	1.0890+03	3.7930+05
18	1.3320+00			1.5200+03	1.5200+03	1.6380+03	0.0	0.0	1.0740+03	1.0740+03	3.8440+05
17	1.2800+00			1.6740+03	1.6740+03	1.8150+03	0.0	0.0	1.0580+03	1.0580+03	3.7980+05
16	1.1770+00			1.6790+03	1.6790+03	1.6770+03	0.0	0.0	1.0300+03	1.0300+03	3.9490+05
15	1.0730+00			-1.0000+02	-1.0000+02	1.8920+03	0.0	0.0	9.8730+02	9.8770+02	3.9970+05
14	9.6950-01			1.6760+03	1.6760+03	1.6770+03	0.0	0.0	9.4060+02	9.4060+02	5.1390+05
13	8.6580-01			-1.0000+02	-1.0000+02	2.1270+03	0.0	0.0	8.9040+02	8.9040+02	4.9620+05
12	7.6200-01			-1.0000+02	-1.0000+02	1.8430+03	0.0	0.0	8.4000+02	8.3980+02	4.9680+05
11	6.5870-01			1.5440+03	1.5440+03	1.6730+03	0.0	0.0	7.9200+02	7.9170+02	5.1820+05
10	5.5560-01			1.2370+03	1.2370+03	1.2880+03	0.0	0.0	7.4830+02	7.4790+02	4.9630+05
9	5.0430-01			1.1010+03	1.1010+03	1.1180+03	0.0	0.0	7.1890+02	7.1850+02	5.2210+05
8	4.5310-01			1.0070+03	1.0070+03	1.0480+03	0.0	0.0	7.0130+02	7.0080+02	5.3450+05
7	4.0190-01			9.6790+02	9.6790+02	1.0400+03	0.0	0.0	6.8550+02	6.8510+02	5.6230+05
6	3.2140-01			8.9360+02	8.9360+02	1.0570+03	0.0	0.0	6.7400+02	6.7360+02	5.7910+05
5	1.6060-01			8.4260+02	8.4260+02	1.0370+03	0.0	0.0	6.6510+02	6.6480+02	6.1290+05
4	0.0			7.7110+02	7.7110+02	8.6480+02	7.6650+02	1.3890+03	6.5820+02	6.5790+02	5.6590+05
3	-1.3330-01	6.5490+02	6.5510+02				7.1390+02	0.0	6.5640+02	6.5620+02	5.8110+05
2	-2.6670-01	6.5430+02	6.5440+02				6.9440+02	0.0	6.5530+02	6.5520+02	0.0
1	-4.0000-01	6.5390+02	6.5390+02				6.8180+02	0.0	6.5450+02	6.5440+02	0.0

Figure 16.9.5: Output Description (Fourth Continuation)

printed from the driver routine LEVDRV (Figure 16.9.5).

Debug printout can be obtained between cycles IBGO and IBSTOP by specifying the input variables in block 51. The amount of debug printout can be increased by increasing the input variable IBNEW from 0 to 4 (acceptable values are 0,1,2,3,4).

16.10 References

16-1. A. M. Tentner and H. U. Wider, "LEVITATE - A Mechanistic Model for the Analysis of Fuel and Cladding Dynamics under LOF Conditions," *Intl. Mtg. on Fast Reactor Safety Technology*, Seattle, WA, August 1979.

16-2. W.R. Bohl and T.J. Heames, Unpublished information, Argonne National Laboratory, 1974.

16-3. W.R. Bohl and M.G. Stevenson, "A Fuel Motion Model for LMFBR Unprotected Loss-of-Flow Accident Analysis," CONF-730414-P2, Ann Arbor, MI, April 1973.

16-4. M.G. Stevenson, et al., "Current Status and Experimental Basis of the SAS LMFBR Accident Analysis Code System," *Proc. of the Fast Reactor Safety Meeting, Beverly Hills*, CA, April 1974.

16-5. H.U. Wider, "PLUTO2: A Computer Code for the Analysis of Overpower Accident in LMFBRs," *Trans. Am. Nucl. Soc.*, vol. 27, p. 533, 1977.

16-6. A.M. Tentner, H.U. Wider and C.H. Bowers, "A Mechanistic Model for Fuel Flow Regimes and Fuel Plateout," *Trans. Am. Nucl. Soc.*, vol. 30, p. 448, 1978.

16-7. M. Epstein and G.M. Hauser, "The Melting of Finite Steel Slabs in Flowing Nuclear Reactor Fuel," *Nucl. Eng. Des.*, vol. 52, 1979.

16-8. B. Spencer, et al., "Results of Fuel Freezing Tests with Simulated CRBR-Type Fuel Pins," *Trans. Am. Nucl. Soc.*, vol. 30, 1978.

16-9. S.J. Hakim, "Analysis of a Prototypic and a Simulant Experiment Related to Freezing in Flow Channels," Specialists' Workshop on Predictive Analysis of Material Dynamics in LMFBR Safety Experiments, Los Alamos Scientific Laboratory, NM, March 1979.

16-10. C.H. Bowers, et al., Unpublished information, Los Alamos Scientific Laboratory, 1979.

16-11. A.M. Tentner and H.U. Wider, "New Aspects in the Analysis of Fuel, Dynamics during Loss-of-Flow Transients," *Trans. Am. Nucl. Soc.*, vol. 38, 1982.

16-12. A.M. Tentner and H.U. Wider, "Pressure Drop Modeling in Variable Area, Multiphase, Transient Flow," in *Multiphase Transport: Fundamentals, Reactor Safety Applications*, N. Veziroglu, Ed., Hemisphere Publishing Co. N.Y., May 1980.

16-13. A.M. Tentner and H.U. Wider, "The Influence of Steel Vapor Pressure on Fuel Motion in Voided LMFBR Channels," *Trans. Am. Nucl. Soc.*, vol. 34, 1980.

16-14. R.E. Henry et al. "Pressure Pulse Propagation in Two-Phase One- and Two-component Mixtures," ANL-7792, Argonne, IL, March 1971.

16.11 Nomenclature

Symbol	Description	Units
$A_{k,i}$	Cross sectional area of component k, in the coolant channel in the axial cell i	m^2
$A_{k,ca,i}$	Cross sectional area of component k, in the fuel-pin cavity, in axial cell i.	m^2
$A_{j,k,i}$	Contact lateral area between component j and component k in the axial cell i	m^2
AXMX	Reference cross-sectional area which determines the volume AXMX $\times \Delta z_i$, over which all component densities are smeared to yield the generalized smeared densities	m^2
AFRV	Input, sodium vapor friction coefficient	
BFRV	Input, exponent of vapor friction coefficient	
C_{AREA}	Coefficient used to define the contact area between two components. Has values between 0 and 1	
$C'_{D,Mi,fu,i}$	Generalized drag coefficient between the gas mixture and the molten fuel in the axial cell i	Kg/m^4
$C'_{in,Mi,fu,i}$	Generalized inertial force coefficient, used in the definition of the inertial force exerted between the gas mixture and the molten fuel in the axial cell i	Kg/m^3
$C_{ff,cl,i}$	Coefficient defining the fraction of the cladding perimeter covered by the solid fuel crust in the axial cell i	
$C_{ff,sr,i}$	Coefficient defining the fraction of the hexcan wall perimeter covered by the solid fuel crust in the axial cell i	
$C_{fu,cl,i}$	Coefficient defining the fraction of the cladding perimeter covered by the molten and frozen fuel in the axial cell i	
$C_{fu,sr,i}$	Coefficient defining the fraction of the hexcan wall covered by the molten and frozen fuel in the axial cell i	
$C_{fu,ff,cl,i}$	Coefficient defining the fraction of the frozen fuel crust on the cladding covered by molten fuel	
$C_{fu,ff,sr,i}$	Coefficient defining the fraction of the frozen fuel crust on the hexcan wall covered by molten fuel	
$C_{mfu,cl,i}$	Coefficient defining the fraction of the cladding perimeter covered by molten fuel only, in cell i	
$C_{mfu,sr,i}$	Coefficient defining the fraction of the hexcan wall covered by molten fuel only, in cell i	
$C_{p,fu}$	Specific heat for the fuel	J/(kg·K)
$C_{se,ff,cl,i}$	Coefficient defining the fraction of the frozen fuel crust on the cladding covered by molten steel	
$C_{se,ff,sr,I}$	Coefficient defining the fraction of the frozen fuel crust on the canwall covered by molten steel	
CMNL	Input, liquid sodium compressibility	Pa^{-1}
CMFU	Input, liquid fuel compressibility	Pa^{-1}
$D_{H,Mi,i}$	Hydraulic diameter for the gas mixture in the coolant channel in cell i	m
$f_{Mi,i}$	Friction coefficient used to characterize the friction between the gas mixture and the channel walls in cell i	

continues on next page

Table 16.11.1 – continued from previous page

Symbol	Description	Units
$F_{area,bd,lu,i}$	Fraction of lateral chunk area in contact with the channel boundaries in cell i	
$h_{k,i}$	Enthalpy of component k in the axial cell i	J/kg
$H_{j,k,i}$	Heat-transfer coefficient between components j and k in cell i.	J/(m^{2}s-K)
$H'_{j,k,i}$	Generalized heat transfer coefficient, similar otherwise to $H_{j,k,i}$ $H'_{j,k,i} = H_{j,k,i} \cdot \frac{A_{j,k,i}}{AXMX \cdot \Delta zi}$	J/(m^{3}s-K)
$h_{Na,liq,i}$	Enthalpy of liquid sodium at saturation in cell i	J/kg
$h_{Na,vap,i}$	Enthalpy of sodium vapor at saturation in cell i	J/kg
$h_{Na,lg,i}$	Heat of vaporization for sodium in cell i	J/kg
$h_{fv,lg,i}$	Heat of vaporization for fuel	J/kg
$k_{Mi,I}$	Conductivity of the sodium and fission-gas mixture in the axial cell i	J/(M-s-K)
$L_{cl,I}$	Perimeter of all pins in a subassembly in the axial cell i	m
$L_{sr,i}$	Same as above, for the canwall	m
$L_{ff,cl,I}$	Perimeter of the cladding fuel crust in a subassembly in the axial cell i	m
$L_{ff,sr,i}$	Same as above, for the canwall fuel crust	m
$l_{ff,cl,i}$	Thickness of the fuel crust on cladding in the axial cell i	m
$l_{ff,sr,i}$	Same as above for the canwall fuel crust	m
$M_{k,i,ic}$	Mass of component k is the axial cell i of one subassembly of channel ic	kg
$M_{k,i,ic}{}^{o}$	Same as above, but referring to the initial conditions	kg
N_{pins}	Number of fuel pins in the subassembly	
$P_{ch,i}$	Total pressure in the coolant channel in the axial cell i	Pa
$P_{k,i}$	Partial pressure of component k in cell i	Pa
$P_{k,sat,i}$	Saturation pressure for component k in the cell i, e.g., $P_{Na,sat,i} = P_{sat}(T_{Na,i})$	Pa
$Q_{k,i}$	Energy source in the component k in cell i	J/(kg-s)
$Q_{cl,i}$	Energy source in the cladding in the axial cell i of one subassembly	J/s
R	Universal gas constant	
R_k	Gas constant specific for the component k, $R_k = \frac{R}{M_k}$ where M_k is the molar mass of the component k.	
$R_{ca,k}$	Radius of cavity in the axial cell k	m
$R_{cl,os,i}$	Outer radius of the cladding in the axial cell i	m
$R_{lu,i}$	Radius of the fuel/steel chunks in cell i	m
Re	Reynolds number	
$T_{k,i}$	Temperature of component k in cell i	K
$u_{k,i}$	Velocity of component k in the coolant channel at the boundary $i - 1/2$	m/s
$v_{Na,i}$	Specific volume of sodium in cell i	m^3/kg
$v_{Na,liq,i}$	Specific volume of the liquid sodium at saturation in cell i	m^3/kg
$v_{Na,vap,i}$	Specific volume of the sodium vapor at saturation in cell i	m^3/kg
$V_{k,i}$	Volume occupied by component k in cell i	m^3
$X_{Na,i}$	Quality of the two-phase sodium in cell i	
z	Axial coordinate	m

continues on next page

Table 16.11.1 – continued from previous page

Symbol	Description	Units
Δz_i	Length of axial cell i	m
$\Delta T_{j,k,i}$	Temperature difference between components j and k in the axial cell i	K
Δt	Time increment	s
$\Delta R_{cl,i}$	Current thickness of the cladding in the axial cell i	m
$\Delta R_{cl}{}^0$	Thickness of the original cladding in cell i	m
$\rho_{k,i}$	Physical density of component k in the cell i.	kg/m^3
$\rho'_{k,i}$	Generalized density of component k in cell i, $\rho'_{k,j} = \rho_{k,i} \cdot \frac{A_{k,i}}{\text{AXMX}}$	Kg/m^3
σ_{fu}	Liquid fuel surface tension	J/m^2
σ_{se}	Liquid steel surface tension	J/m^2
$\theta_{k,i}$	Volume fraction of component k in the axial cell i $\theta_{k,i} = \frac{Q_{K,i}}{\text{AXMX}}$	
Subscripts		
bd	Refers to a quantity defined at the cell boundary	
ca	Cavity	
ch	Coolant channel	
cl	Cladding	
ch,op	Open channel. The open channel at any axial location can become smaller than the original coolant channel due to fuel and steel freezing. It can also increase due to the ablation of the cladding and hexcan wall and to the disruption of the fuel pins	
ff	Frozen fuel	
ffc	Frozen fuel on cladding	
ffs	Frozen fuel on structure	
fi	Fission gas	
fu	Molten fuel	
fv	Fuel vapor	
fl	Fuel chunks	
i	Cell number, on the channel grid	
is	Inner cell of cladding or structure	
IN	Inertial	
J	Other components in the coolant channel, exchanging mass, momentum or energy with component k	
k	Component subscript, referring to the component being described. Also used as cell number on the fuel-pin grid	
liq	Indicates a physical property for the saturated liquid, e.g., $h_{Na,liq,i}$	
lq	Indicates a physical property at liquidus	
lu	Solid fuel/steel chunks	
lv	Indicates a physical property that characterizes the transition from liquid to gas, e.g. h_{lv}	
Mi	Mixture of sodium vapor, fission gas, fuel vapor, steel vapor, and liquid sodium	
Mo	Momentum	
Na	Sodium	
Nl	Liquid sodium	
Nv	Sodium vapor	

continues on next page

Table 16.11.1 – continued from previous page

Symbol	Description	Units
Os	Outer cell of cladding or structure. The "outer" surface is always the surface facing the coolant channel	
pin	Fuel pin	
sat	Indicates a physical property at saturation, e.g., T_{sat} ($P_{k,i}$)	
se	Molten steel	
sl	Steel chunks	
so	Indicates a physical property at solidus	
sr	Structure, i.e., hexcan wall or wall of the experimental loop	
st	Refers to a stationary component, such as the solid fuel in the pin	
sv	Steel vapor	
vap	Indicates a physical property for the saturated dry vapor, e.g., $h_{Na,vap,i}$	
vg	Vapor/gas mixture	
Superscripts		
cond	Condensation	
L	Used to indicate a lateral area, as opposed to a cross-sectional area	
N	The value of any quantity at the beginning of the current time step	
$n+1$	The value of any quantity at the end of the current time step. If no superscript is present, the default is n	
N	Used in relation to the cylindrical chunks, to indicate the area of the lower and upper base	
vap	Vaporization	



Nuclear Science and Engineering Division

Argonne National Laboratory
9700 South Cass Avenue
Lemont, IL 60439

www.anl.gov



Argonne National Laboratory is a U.S. Department of Energy laboratory managed by UChicago Argonne, LLC.