
MELPROG-PWR/MOD1: A Two-Dimensional, Mechanistic Code for Analysis of Reactor Core Melt Progression and Vessel Attack Under Severe Accident Conditions

Manuscript Completed: January 1989

Date Published: May 1989

Edited by:
S.S. Dosanjh

Contributors:

J.L. Tomkins, M.F. Young, J.F. Dearing,* R.C. Smith,
S.S. Dosanjh, J.E. Kelly, A.J. Grimley, C.J. Wheatley,**
P.J. Maudlin,* T.J. Heames,*** W.J. Camp

Sandia National Laboratories
Albuquerque, NM 87185

*Los Alamos National Laboratory

**United Kingdom Atomic Energy Authority

***Science Applications International Corp.

Prepared for
Division of Systems Research
Office of Nuclear Regulatory Research
U.S. Nuclear Regulatory Commission
Washington, DC 20555
NRC FIN A1342

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.

JHP
DISTRIBUTION OF THIS DOCUMENT IS UNLIMITED
MASTER

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.

DISCLAIMER

Portions of this document may be illegible in electronic image products. Images are produced from the best available original document.

ABSTRACT

Since the Three Mile Island accident, increased emphasis has been placed on the analysis of severe reactor accidents because such accidents represent the dominant public risk from nuclear reactors. The U.S. Nuclear Regulatory Commission has made the development of mechanistic models for severe accident progression a major priority. The purpose of these models is to provide detailed, best-estimate, coupled analyses of all the major phenomena involved in the reactor vessel and coolant system in the course of the accident. To meet this objective, the MELPROG computer code is being developed. This report describes the two-dimensional, pressurized water reactor (PWR) version of the MELPROG computer code, MELPROG/PWR-MOD1. A boiling water reactor (BWR) version of the code, MELPROG/BWR-MOD1, is being developed and will be described in future reports. However, preliminary BWR work is described in this report.

The purpose of MELPROG is to provide for an accident sequence a description of (1) the state of the reactor core and surrounding in-vessel environment as a function of time, (2) the disposition of core materials (in particular, fission products) contained within the reactor coolant system boundary as a function of time, and (3) the state of the debris that enters the containment if the reactor vessel fails. MELPROG is coupled to the TRAC-PF1 RCS thermal-hydraulics code to provide an integrated analysis of the behavior of core, vessel, and reactor coolant systems during severe accidents. MELPROG treats core degradation and loss of geometry, debris formation, core melting, attack on supporting structures, slumping, melt/water interactions and vessel failure. The key element in MELPROG is the use of detailed modeling for the entire damage progression and failure sequence. Emphasis is also placed on the rates of hydrogen, steam and fission product formation, and transport to containment during the accident.

TABLE OF CONTENTS

	<u>Page</u>
1.0 INTRODUCTION AND CODE DESIGN	1-1
1.1 MELPROG/TRAC Structure and Approach	1-3
1.2 Core Melt Phenomenology	1-6
1.3 The Present Report	1-10
1.4 References	1-11
2.0 FLUIDS MODULE	2-1
2.1 Basic Equations	2-2
2.2 Solution Algorithm	2-4
2.2.1 Momentum Equations	2-5
2.2.2 Basic Step	2-8
2.2.3 Mass and Energy Stabilizers	2-11
2.3 Imbedded Paths	2-11
2.4 Methods for Solving Systems of Equations	2-12
2.5 Logic for Set Switching	2-13
2.6 Time Step Control	2-14
2.7 Equation of State	2-15
2.8 Auxiliary Equations Required for Closure	2-18
2.8.1 Flow Regime Maps	2-19
2.8.2 Structure Heat Transfer and Drag	2-23
2.8.3 Corium Heat Transfer	2-23
2.9 Module Interfaces and Boundary Conditions	2-28
2.9.1 Stand-Alone Boundary Conditions	2-28
2.9.2 Interfaces with Other Modules of MELPROG	2-29
2.9.3 Link to TRAC	2-30
2.9.3.1 Model Limitations	2-30
2.9.3.2 Calculation Control	2-33
2.9.4 Link to VICTORIA	2-35
2.10 FLUIDS Nomenclature	2-35
2.11 References	2-36

TABLE OF CONTENTS

(continued)

	<u>Page</u>
3.0 CORE MODULE	3-1
3.1 General Description	3-2
3.2 Detailed Description of Core	3-4
3.2.1 Geometry	3-4
3.2.2 Core Heat Transfer	3-9
3.2.2.1 Conduction Heat Transfer	3-9
3.2.2.2 Phase Transition Heat Transfer	3-17
3.2.2.3 Conduction Limited Freezing	3-20
3.2.3 Chemistry	3-26
3.2.3.1 Oxidation Reactions	3-26
3.2.3.2 Eutectic Reactions	3-29
3.2.4 Energy Generation	3-29
3.2.5 Fuel Rod Mechanics	3-31
3.2.6 Candling	3-31
3.2.6.1 Problem Definition	3-32
3.2.6.2 Solution	3-33
3.2.6.3 Iteration Strategy	3-40
3.2.6.4 Conservation Equations	3-41
3.2.6.5 Momentum Equation	3-42
3.2.6.6 Mass and Energy Equations	3-45
3.2.6.7 Pressure Equation	3-45
3.2.6.8 Incremental CRMESH and FLUIDS Terms	3-46
3.2.7 Dynamic Mesh Treatment	3-47
3.2.7.1 General Remeshing Considerations	3-47
3.2.7.2 Material Region Change Terms	3-48
3.2.7.3 Material Region Creation	3-52
3.2.7.4 Combining and Deleting Material Regions	3-58
3.2.7.5 Subdividing and Combining Heat Transfer Nodes	3-59

TABLE OF CONTENTS

(continued)

	<u>Page</u>
3.3 References	3-61
4.0 STRUCTURES MODULE	4-1
4.1 Structure Heat Transfer	4-1
4.2 Structure Types	4-1
4.2.1 Wall Type Structures	4-3
4.2.2 Plate Type Structures	4-5
4.2.3 Column Type Structures	4-5
4.2.4 General Type Structures	4-6
4.3 Mesh Arrangement	4-6
4.4 Material Properties	4-6
4.5 Conduction Heat Transfer Solution Technique	4-7
4.5.1 Formulation for Material Nodes	4-7
4.5.2 Surface Nodes	4-8
4.5.3 Matrix Solution	4-9
4.6 Phase Transition Heat Transfer	4-10
4.7 Structure Oxidation	4-13
4.8 Structural Mechanics Modeling	4-13
4.9 Modeling Design Considerations	4-14
4.10 Mechanics Modeling	4-15
4.10.1 Lower Core Grid Plate	4-17
4.10.2 Elliptic Plate	4-20
4.10.3 Cylinder	4-22
4.10.4 Columns	4-23
4.10.5 Bolts	4-24
4.10.6 Joints	4-24
4.10.7 Eggcrate Grill	4-24
4.10.8 Example	4-26
4.11 References	4-26
5.0 RADIATION MODULE	5-1
5.1 Model Theory and Design	5-1

TABLE OF CONTENTS
(continued)

	<u>Page</u>
5.2 Radiation Heat Transfer Model Implementation	5-1
5.2.1 Geometry and Fluid Fields Modeled	5-5
5.2.2 Solution Procedure	5-5
5.2.3 View Factor Calculation	5-8
5.2.4 Material Properties and Mean Transmission Coefficients	5-11
5.2.5 Summary of Basic Modeling Assumptions	5-12
5.2.6 RADIATION Module Interaction with Other Modules	5-12
5.3 References	5-12
6.0 DEBRIS MODULE	6-1
6.1 Calculation Sequence	6-2
6.2 Debris Bed Formation and Release Model	6-4
6.3 Post-Dryout Debris Bed Model	6-6
6.3.1 Heat Transfer	6-6
6.3.2 Melt Relocation Model	6-8
6.3.3 Phase Diagrams	6-10
6.3.4 Zr Oxidation Model	6-11
6.4 Future Work	6-12
6.4.1 Conservation of Mass	6-12
6.4.2 Liquid Motion	6-13
6.4.3 Solid Motion	6-15
6.4.4 Conservation of Energy	6-15
6.4.5 Phase Diagrams	6-16
6.4.6 Closure	6-18
6.5 References	6-20
7.0 VICTORIA MODULE	7-1
7.1 Methodology	7-3
7.1.1 Solution Domain	7-3

TABLE OF CONTENTS

(continued)

	<u>Page</u>
7.1.2 Modeling Overview	7-5
7.1.3 Calculational Flow	7-6
 7.2 Species Behavior and Transport Within the Fuel Geometry	 7-11
7.2.1 Governing Equations	7-13
7.2.2 Intragranular Processes	7-14
7.2.2.1 Equilibrium Chemistry	7-14
7.2.2.2 Solid-State Diffusion	7-17
7.2.3 Open Porosity Processes	7-18
7.2.3.1 Chemical Equilibrium	7-19
7.2.3.2 Diffusive Transport	7-19
7.2.4 Ex-Pellet Processes	7-21
7.2.4.1 Equilibrium Chemistry	7-21
7.2.4.2 Transport	7-22
 7.3 Bulk Gas Species Continuity	 7-22
7.3.1 Species Conservation Equation	7-22
7.3.2 Numerical Scheme	7-24
7.3.3 Information for the Code User	7-27
 7.4 Chemistry	 7-27
7.4.1 Introduction	7-27
7.4.2 Equilibrium Chemistry	7-31
7.4.2.1 Derivation of the Equations	7-31
7.4.2.2 Application to the Gas Region	7-36
7.4.2.3 Application to the Structure Surface Region	7-39
7.4.2.4 Application to the Clad/Gap Interface	7-40

TABLE OF CONTENTS
(continued)

	<u>Page</u>
7.4.2.5 Application to the Fuel	7-40
7.4.2.6 The Solution Method	7-41
7.4.2.7 Subroutine Design	7-44
7.4.3 Species	7-45
7.5 Radioactive Decay Heating	7-50
7.5.1 Volumetric and Surface Decay Powers	7-51
7.5.2 Information for the Code User	7-52
7.6 Species Aerosol Behavior and Transport	7-52
7.6.1 Description of the Model	7-54
7.6.1.1 Gas Properties	7-55
7.6.1.2 Flow Properties	7-57
7.6.1.3 Initial and Source Distributions	7-59
7.6.1.4 Particle Mobilities	7-60
7.6.1.5 Deposition Rates	7-61
7.6.1.6 Agglomeration Rates	7-64
7.6.2 Numerical Method	7-67
7.6.2.1 Discretization Method	7-67
7.6.2.2 Solution Method	7-70
7.6.2.3 Moments of the Discretized Distribution	7-71
7.7 References	7-72
8.0 IFCI MODULE	8-1
8.1 The FCI Process	8-2
8.2 Previous FCI Predictive Models	8-3
8.3 A Predictive Model for FCIs	8-4
8.4 IFCI in a MELPROG Context	8-4
8.5 IFCI Code Description	8-5

TABLE OF CONTENTS

(continued)

	<u>Page</u>
8.5.1 General Code Structure	8-5
8.5.2 Fragmentation Model	8-6
8.5.3 Surface Area Transport	8-12
8.6 FCI Example Problem	8-12
8.6.1 Problem Setup	8-13
8.6.2 Coarse Mixing Phase	8-15
8.7 Triggering and Propagation Phase	8-20
8.7.1 Triggering	8-20
8.7.2 Propagation	8-20
8.8 Discussion of Results	8-26
8.9 Future Work	8-27
8.10 References	8-27
9.0 EJECT MODULE	9-1
9.1 Hole Ablation	9-1
9.2 Gas Blowthrough	9-3
9.3 Jet Disruption	9-5
9.4 References	9-8
10.0 MATERIAL PROPERTIES	10-1
10.1 Detailed Description	10-1
10.2 References	10-11
APPENDIX A	A-1
A.0 CODE STRUCTURE	A-1
A.1 FLUIDS Module	A-1
A.2 CORE Module	A-5
A.3 STRUCTURES Module	A-7

TABLE OF CONTENTS

(continued)

	<u>Page</u>
A.3.1 Heat Transfer Routines	A-8
A.3.2 Mechanics Routines	A-10
A.3.3 Flow Areas and Hydraulic Diameters	A-15
A.4 DEBRIS Module	A-15
A.5 RADIATION Module	A-17
A.6 TRAC/MELPROG Link Computational Flow	A-19
A.7 MELPROG Data Input and Initialization Module (INPUT)	A-21
A.8 Data Management Structure Used in MELPROG	A-22
 APPENDIX B	 B-1
B.0 INPUT INSTRUCTIONS	B-1
B.1 MOD1 Input Instructions	B-1
B.1.1 General Input	B-1
B.1.2 FLUIDS Module Input	B-3
B.1.2.1 Scalar Data	B-3
B.1.2.2 Array Data	B-3
B.1.2.3 Boundary Conditions	B-4
B.1.2.4 Initial Conditions	B-5
B.1.2.5 Geometry Data	B-6
B.1.3 CORE Module Input	B-6
B.1.3.1 Scalar Data and Power Shape	B-6
B.1.3.2 Model Definition Data	B-7
B.1.3.3 Model Location Data	B-8
B.1.4 STRUCTURE Module Input	B-8
B.1.4.1 Scalar Data	B-8
B.1.4.2 Model Definition and Boundary Condition	B-9
B.1.4.3 Model Location and Geometry Data	B-9
B.1.5 RADIATION Module Input	B-11

TABLE OF CONTENTS

(continued)

	<u>Page</u>
B.1.6 DEBRIS Module Input	B-12
B.1.7 Input for the Mechanics Calculation (at initialization only)	B-12
 APPENDIX C	 C-1
C.0 GRAPHICS OUTPUT	C-1
C.1 CMOUT Graphics File Structure	C-1
C.2 TRAP Usage on CTSS	C-3
 APPENDIX D	
D.0 HEAT TRANSFER AND FLOW CORRELATIONS	D-1
D.1 Wall Shear	D-1
D.2 Interfacial Heat Transfer and Shear	D-3
D.3 Wall-to-Fluid Heat Transfer	D-8
D.3.1 Wall-to-Fluid HTC Selection Logic	D-8
D.3.2 HTC Correlations	D-11
D.3.2.1 Single-Phase Liquid (Heat- Transfer Regimes 1 and 12)	D-12
D.3.2.2 Nucleate-Boiling HTCs (Heat Transfer Regime 2)	D-13
D.3.2.3 Transition-Boiling HTCs (Heat Transfer Regime 3)	D-15
D.3.2.4 Film-Boiling HTCs (Heat Transfer Regime 4)	D-16
D.3.2.5 Single-Phase Vapor HTCs (Heat Transfer Regime 6)	D-18
D.3.2.6 Condensation HTCs (Heat Transfer Regime 11)	D-19
D.3.3 Critical Heat Flux (CHF)	D-19
D.3.4 Minimum Stable Film-Boiling Temperature, T _{min}	D-21
D.4 References	D-22

LIST OF FIGURES

<u>Figure</u>	<u>Page</u>
2.1 Flow-regime map for two-phase vertical flow.	2-20
2.2 Flow-regime map for two-phase horizontal flow based on work by Baker.	2-22
2.3 Boiling curve showing the liquid and vapor heat transfer coefficients as functions of temperature.	2-24
2.4 Liquid and vapor heat transfer coefficients as a function of vapor fraction in the film boiling regime.	2-25
2.5 Map showing where the different heat transfer coefficients are applicable as a function of temperature and alpha.	2-26
3.1 Typical fuel rod model.	3-5
3.2 Typical fuel rod model with grid spacers.	3-7
3.3 Fuel rod used in an out-of-pile experiment.	3-8
3.4 BWR control rod assembly.	3-10
3.5 BWR control blade model.	3-11
3.6 Node arrangement for a fuel rod.	3-13
3.7 Geometry at time t , prior to the formation of a crust.	3-22
3.8 Geometry at time $t + \Delta t$.	3-22
3.9 Flow chart of the logic in the eutectic formation subroutine, CREUTK.	3-30
3.10a Flow chart showing structure of the CANDLE subroutine.	3-38
3.10b Flow chart showing structure of the CANDLE subroutine. (continued)	3-39
3.11 Single node region melting logic.	3-50
4.1 Typical plate and wall type structures.	4-2
4.2 Structure arrangement in a FLUIDS computational cell.	4-4
4.3 Westinghouse PWR schematic.	4-16

LIST OF FIGURES
(continued)

	<u>Page</u>
4.4 Cell of a Westinghouse lower core grid plate, top view.	4-18
4.5 Eggcrate grill section.	4-25
5.1 Typical geometries treated in the RADIATION module.	5-6
6.1 Zr-Fe phase diagram given by Hansen.	6-17
6.2 Possible melt compositions for a given local (solid and liquid) composition.	6-19
7.1 Simple mesh cell diagram dividing the reactor vessel into six axial levels and four radial rings.	7-4
7.2 Schematic of the relevant geometry and phenomena related to species behavior and transport, which are modeled within each gross two-dimensional mesh cell.	7-7
7.3 The program structure of the VICTORIA module.	7-9
7.4 Flow chart of the logic flow in subroutine VIKI.	7-10
7.5 Sample of the type of mesh used in the VICTORIA diffusion calculation.	7-12
7.6 Spatial indexing scheme shown for a single bulk gas cell.	7-25
7.7 Flow chart of the logic used in subroutine VGTRAN to solve the bulk gas species continuity equations.	7-28
7.8 Flow chart of the logic flow in the DECAY subroutine.	7-53
8.1 Finite difference mesh cell and variable location in IFCI.	8-9
8.2 IFCI test problem mesh setup.	8-14
8.3 Melt outline in FITS experiment.	8-16
8.4 Melt volume fraction at time 0.02 s.	8-17
8.5 Melt volume fraction at time 0.2 s.	8-18
8.6 Melt volume fraction time at 0.3 s.	8-19
8.7 Melt diameter versus radius and height at time 0.31 s.	8-22

LIST OF FIGURES
(continued)

	<u>Page</u>
8.8 Pressure versus radius and height at time 0.304 s.	8-23
8.9 Melt size distributions from IFCI problem and FITSOD posttest debris.	8-25
10.1a Flow chart showing structure of the MELPRO subroutine.	10-4
10.1b Flow chart showing structure of the MELPRO subroutine (continued).	10-5
10.1c Flow chart showing structure of the MELPRO subroutine (continued).	10-6
A.1 Overall MELPROG structure.	A-2
A.2 Functional diagram of FLUIDS module routines.	A-3
A.3 Functional diagram of CORE module routines.	A-6
A.4 Functional diagram of structure heat transfer routines.	A-9
A.5 Functional diagram of structure mechanics routines.	A-11
A.6 Functional diagram of DEBRIS module routines.	A-16
A.7 Functional diagram of RADIATION module subroutines.	A-18
A.8 Outer iteration flow chart for TRAC/MELPROG.	A-20
A.9 Functional diagram of INPUT module subroutines.	A-22
D.1 Boiling curve.	D-9

LIST OF TABLES

	<u>Page</u>
2.1 Fields in the Fluid Dynamics Calculation.	2-2
2.2 TRAC Water Equation of State Limits.	2-16
2.3 Definition of the IENTRY Flag.	2-34
3.1 Heat Transfer Node Characteristics.	3-12
3.2 Parameters for Parabolic Kinetics Equations. (Zr oxidation)	3-27
3.3 Parameters for the Parabolic Kinetics Equations. (steel oxidation)	3-28
3.4 CANDLE Working Array Descriptions.	3-34
3.5 Conservation Equation Symbols Used in the CANDLE Subroutine.	3-44
3.6 Material Region Characteristics.	3-48
3.7 Hierarchial Order of the Form of Change for Each Surface.	3-53
3.8 Dimensional Change Options.	3-55
3.9 Dimensional Change Method Used for the Various Physical Situations.	3-56
4.1 Examples of Structure Types.	4-3
5.1 View Factor Matrix for a Typical Type 1 Cell.	5-10
6.1 Summary of the DEBRIS Phase Change Rules.	6-11
7.1 VICTORIA Species Set.	7-46
7.2 VICTORIA Element Set.	7-48
7.3 Free Energy Data for the 99 Species in VICTORIA.	7-48
7.4 Elements to be Added.	7-50
8.1 Length Scales from the IFCI and FITS "D" Series Experiments.	8-24
10.1 Materials Treated in the MELPRO Material Properties Routine.	10-2

LIST OF TABLES
(continued)

	<u>Page</u>
10.2 Properties Provided by the MELPRO Routine.	10-3
10.3 Rules for Determining Mixture Compositions.	10-8
A.1 Ring Load Table Data Structure.	A-12
A.2 Geometry Table Data Structure.	A-23
A.3 Loading Table Data Structure.	A-24
C.1 CMOUT File Structure.	C-2
C.2 Structure of P Block Header.	C-2
C.3 Structure of the Q Block Header.	C-4
C.4 Structure of LCDATA Block for CMOUT.	C-5
C.5 Structure of the Geometry Block in CMOUT.	C-6
C.6 Structure of the Time Edit Block.	C-7
D.1 TRAC-PF1/MOD1 Heat Transfer Regimes.	D-11

1.0 INTRODUCTION AND CODE DESIGN

It has been established that the major risk to the public arising from the operation of nuclear power reactors is due to the potential occurrence of severe accidents. Since the Reactor Safety Study [1.1], there has been increasing recognition of the desirability, indeed the necessity, of obtaining in-depth and quantitative understanding of the course and outcome of such accidents. The accident at Three Mile Island (TMI) emphatically underscored that necessity. The U.S.N.R.C. has undertaken a broad-ranging research program to respond to the perceived needs in these areas. On one hand, there has been increased emphasis on more sophisticated methodologies for probabilistic risk assessment (PRA) including the development of semimechanistic computer models for overall accident progression [1.2,1.3]. These models are particularly useful for the development of central estimates of risk and for evaluation of the current uncertainties in that risk due to uncertainties in the outcome of major accident processes (e.g., melt-water interactions, hydrogen combustion, direct containment heating, etc.) On the other hand, in-depth experimental programs are being pursued, which cover many phenomenological areas of importance to severe accidents. Finally, best-estimate, detailed computer models [1.4-1.6] are being developed, which track the course of an accident at a far greater level of detail than is feasible with the computer models associated with PRA uses. These detailed models attempt to incorporate the phenomenology uncovered in related experimental programs and to use state-of-the-art physical and chemical models; they have the goal of being validated against integral and separate effects experimental simulations. They provide the ability to give our best current estimate of the course, outcome, and consequences of a given accident in a given plant. They also allow delineation of the uncertainty in major accident processes caused by uncertainties in the modeling of underlying physical phenomena. These results are intended to feed directly into the uncertainty bands employed in the semimechanistic codes used in PRA applications. Finally, they are designed with sufficiently flexible frameworks as to evolve with a growing data base and changing phenomenological information.

In this report we describe the development of one such best-estimate model, the MELPROG/TRAC computer code [1.5], in terms of the goals and philosophy of the MELPROG project and the underlying code structure and modeling approach. In capsule form those goals are to follow the behavior of coolant, coolant vapor,

hydrogen, core and structural materials and all major fission products--whether in the fuel, as vapors, on aerosols or deposited on structures--throughout the course of a severe accident in the vessel and reactor coolant system. The major calculational products include the rate, location, and physical state of steam, hydrogen, fission products, core and structural materials released to the containment building from the vessel and reactor coolant system. As important as the quantitative predictions of the code is its capability to describe qualitatively the accident sequence as it progresses through a series of damage states and to identify the determining phenomena in deciding the ultimate outcome of the accident. Because of this, the computer code is particularly well-suited for the evaluation of the effects of operator actions, accident management schemes, and "atypical" behavior of engineered systems during the course of the accident. Examples of the latter include failure of the pressure-operated relief valves (PORVs) to re-seat during a station blackout (as occurred in the TMI-2 accident) and small breaks caused by failure of pump seals during a station blackout.

We note that although the MELPROG project is in the advanced stages of development and testing, the exercise of the code has made several notable contributions to our understanding of severe accident melt progression. First of all, the application of the code to high-pressure PWR station blackout accidents has demonstrated the crucial role of natural circulation flows in the steam/hydrogen atmosphere during and following boil-off of coolant from the core region. Such flows are predicted mechanistically by the multidimensional fluids model of MELPROG. Under these high pressure conditions, these flows provide enormous enhancement of heat transfer away from the reactor core to the upper vessel and hot leg of the reactor coolant system. This enhancement of heat transfer leads to a strong potential for early failure of the primary system boundary in the region of the hot leg nozzle or inlet to the pressurizer. Such a failure could lead to rapid depressurization of the reactor prior to onset of severe fuel damage, thus changing the entire risk profile of the accident [1.7-1.9].

MELPROG calculations have demonstrated the strong dependence of core melt temperature, melt fraction, and melt composition on the nature of early-phase rod degradation processes, leading to the definition of experiments to delineate these processes. In addition, MELPROG calculations have been used to study the rate

of hydrogen production and quantity of hydrogen produced during such accidents, thus providing the source term for calculations of hydrogen combustion in containments. The capability of MELPROG to provide the conditions of melt ejection from the vessel during high pressure blowdowns is currently being developed. These calculations bear critically on the important question of containment failure due to direct containment heating.

Finally, we note that because of the code's role in determining our best estimates of risk-dominant accidents, it is being subjected to an extensive analytical and experimental evaluation program. Part of that program involves the use of MELPROG to analyze in-pile fuel degradation experiments as a means of validating role predictions. We note in particular that detailed MELPROG calculations have already been carried out for one such experiment, the DF-1 experiment performed at Sandia National Laboratories [1.10,1.11]. These calculations have demonstrated quantitative agreement between measured and predicted temperatures and end-state conditions (e.g., the location and composition of the blockage). Additional comparisons of MELPROG calculations with appropriate experimental data are currently under way.

1.1 MELPROG/TRAC Structure and Approach

It is useful to summarize briefly some features of the MELPROG/TRAC modeling structure. MELPROG is a modular system of explicitly linked subcodes that follow the behavior of all functional units of the reactor vessel and internals during the accident. Each module is a quasi-independent code that is capable of running as a stand-alone model given suitable input and (time-dependent) boundary conditions. Modules only communicate with each other in an explicit fashion through a driver routine. MELPROG itself is capable of running as a stand-alone model with suitable containment and RCS boundary conditions. Alternately, MELPROG replaces the vessel component of the TRAC-PF1 model, in which case the two codes communicate via a numerically implicit link between the TRAC and MELPROG FLUIDS modules.

The subcodes in MELPROG are:

1. The FLUIDS module that treats multicomponent, multifield compressible flow and heat transfer within the reactor

vessel. In this version, FLUIDS uses a two-dimensional model and treats four fields: liquid coolant, vapors, solid and molten corium. An energy and momentum equation is solved for each field and mass conservation equations are included for each species of interest. In addition, the transport of fission products through the vessel is calculated in the FLUIDS module. (Note that in this report the terms "corium," "debris," and "rubble" are used interchangeably.)

2. The CORE module that treats all structures within the reactor core. The core structures include:

- PWR fuel rods (one or more average rods per ring),
- BWR fuel rods (several representative rods per bundle),
- PWR control rods,
- PWR and BWR poison rods,
- BWR control blades,
- BWR can walls,
- BWR dummy rods, and
- Grid spacers.

The module treats energy generation by decay heat and chemical reactions, heat transfer, thermochemistry, liquefaction and melting of fuel, cladding, and structures, mechanical response and failure of various structures, candling of liquefied materials, freezing and crusting of those materials, as well as oxidation of metallic components and fuel.

3. The STRUCTURES module that treats mechanical and thermal behavior of all vessel structures exterior to the core, as well as the vessel itself. A number of general structure types are used, such as perforated plates, columns, cylindrical shells, etc., to represent specific structures. The structure package currently accommodates PWR's (e.g., Babcock and Wilcox, Westinghouse, and Combustion Engineering). Structural heat transfer is via one- and two-dimensional conduction as appropriate, convection, and phase-change (ablation models). Radiation to and from structures is handled by RADIATION. Ablation mass transport from structures as well as simultaneous crusting and ablation is considered.

4. The RADIATION module that treats radiation heat transfer within the core and vessel. The model uses two-dimensional cells (R-Z) which are diffuse emitters and absorbers. Within each cell, rods, debris, and structures are treated by three-dimensional view factors which change to accommodate geometry changes.
5. The DEBRIS that treats the formation of rubble or debris beds throughout the vessel, using simplified models. Once a rubble bed is formed, DEBRIS handles heat transfer within the bed. However, FLUIDS continues to treat coolant boiling, coolant dryout, reflooding, and quenching. For a dried-out bed, DEBRIS calculates material liquefaction and relocation, refreezing and crust/blockage formation. DEBRIS determines heat fluxes to underlying structure and heat transfer from debris to the FLUIDS coolant field as well.
6. The VICTORIA that treats release and transport of fission products within the vessel. The ultimate driving forces for release of fission products are the differences between equilibrium and existing partial vapor pressures of all relevant molecular species in the overlying vapor. A number of limitations on release are modeled including diffusion within the fuel/cladding matrix, vaporization kinetics and transport across boundary layers. Condensation onto fuel, cladding, structures and preexisting aerosols is treated. The aerosol formation, agglomeration, and settling processes are also modeled.

In addition to treating fission products, VICTORIA also provides for the calculation of vaporization and aerosolization of nonradioactive core materials, e.g., tin alloying agents in the clad, silver and cadmium from control rods, and metaboric acid from poison rods. These materials are treated on an equal chemical and physical footing with radioactive fission products.

7. The IFCI (integrated fuel coolant interaction) model that treats explosive and nonexplosive melt-water interactions, including detailed modeling of the detonation process itself. IFCI uses correlational models for fuel/coolant intermixing, film boiling, and fuel fragmentation. These models are used in a finely

noded FLUIDS treatment to calculate detonation wave propagation, vapor generation and post-explosion expansion.

8. The EJECT module that analyzes the release of materials from the vessel associated with various vessel failure modes. EJECT will handle the entire spectrum of release modes from gravity slumps to high pressure blowdowns. For the latter, EJECT will calculate melt fragmentation by effervescence, atomization and hydrodynamic fragmentation. It will also calculate the increase in failure size by ablation during the blowdown process.

Detailed discussions of the above modules are provided in Sections 2 through 9. The approach in building MELPROG/TRAC is that every module represents or attempts to extend the state of the art in system-level modeling of all important phenomena with which it deals. The models employed are as mechanistic as possible given the size and complexity of the system and the limits of computational resources. Finally, each module is structured so as to allow evolutionary upgrading of models as new information becomes available. Revolutionary upgrades are similarly allowed for; the modular structure of MELPROG allows the complete removal and replacement of an entire module with minimal impact on the remainder of the code. Indeed in the MELPROG/MOD1 version, the one-dimensional, three-velocity FLUIDS module and the simple PINS module (for core behavior) of MELPROG/MOD0 were replaced by a two-dimensional, four-velocity FLUIDS module and an extremely detailed CORE module which has redefined the state of the art for core degradation modeling. Similarly the DEBRIS module of MOD0 is currently being replaced in MOD1 by a detailed multimaterial, two-dimensional, porous medium meltdown model.

1.2 Core Melt Phenomenology

We have noted above some of the physical and chemical processes modeled by MELPROG. However, such a list, by itself, does not provide much insight into how these phenomena enter the accident sequence. A useful overview of the phenomenology can be had by briefly describing the chronology of a typical accident--as simulated by MELPROG.

In a typical accident simulation, the calculation begins with an initially intact core and vessel. A break in the coolant system or loss of heat removal capability is specified. In either case, coolant loss from the system is calculated by the FLUIDS module. In a complete system simulation, TRAC-LINK is employed to couple the vessel with a TRAC-PF1 model of the RCS. In some cases, a set of boundary conditions is used to simulate the TRAC calculation. This latter approach is most appropriate if a detailed TRAC or RELAP calculation of the coolant blowdown or boiloff sequence is available.

As the coolant level drops below the top of the core, rapid core heatup begins due initially to decay heat generated in the fuel rods. During this entire heatup stage, multidimensional convection in the core and above-core region is an important means of transferring heat from the core to the upper plenum structures and potentially to the RCS. When temperatures surpass about 1273 K, rapid oxidation of the zircaloy cladding by hot steam begins near the top of the core. This deposits even more energy into the core, resulting in a more rapid temperature increase. Additionally, large quantities of hydrogen gas are generated. During these later stages, radiation heat transfer becomes important and ultimately dominant as a heat transfer mechanism. In some cases, ballooning of the cladding will occur. A "zircaloy burn front" can develop in the upper regions of the core and move upstream to the lower portions of the core as the accident progresses. The upper regions of the core are often sufficiently hot and damaged to fail even before water has completely boiled out of the core region. The heatup of the core, cladding oxidation, and rod failure are all calculated by the CORE module in concert with the FLUIDS module.

While the core is heating up, heat is transported to surrounding structures by FLUIDS (convection heat transfer) and by the RADIATION module. Both the core barrel and the upper core grid plate heat up rapidly. The STRUCTURES module calculates the thermal and mechanical response to these loads. For some accidents, the upper grid plate is calculated by STRUCTURES to fail early in the accident.

During the heatup process, the first failures in the core are typically the control rods. As these fail, they form debris which is passed by CORE to the FLUIDS module for transport and possible formation of debris beds. As rods begin to fail, CORE also adds this debris to the FLUIDS corium field. A simple

debris bed formation model is employed to form stationary debris beds in the lower core region (as well as in the lower head). When debris beds are formed, FLUIDS passes control of the corium involved to the DEBRIS module.

Ordinarily, the debris will hang up on lower pin stubs in narrowed or blocked sections where candled cladding has refrozen. If water remains in the region where debris beds form, FLUIDS maintains control of the heat transfer from the debris, and performs an average calculation for each fluid node. However, if the region is initially dry or becomes dry later, DEBRIS will perform a finely meshed analysis of debris heatup, melting, relocation, crusting, and crust melting. If a debris bed begins to move through failure of its supporting structure, then DEBRIS releases the debris bed and returns control to the FLUIDS module.

Eventually, the debris bed becomes largely molten and the thermal loads on the grid plate can become sufficient to cause local failures of the grid plate. This failure process as well as thermal processes in the structures are modeled on a cell-by-cell basis by STRUCTURES. If the grid plate fails, the partially molten corium slumps into the lower head (under control of FLUIDS) and is quenched. During such quenching processes, copious quantities of steam and hydrogen are generated. These are transported by FLUIDS. Note that a steam explosion may occur at the point of fuel slumping into the water pool. If such an explosion occurs, significant shock and impulse loads can be delivered to the vessel. The code calculates such loads and assesses the potential for vessel or RCS failure under the estimated loads.

Finally, the quenched debris heats up, boils the remaining coolant dry, and remelts under renewed control of DEBRIS. The molten corium melts through the vessel wall. This process is again calculated by STRUCTURES using the thermal fluxes generated by DEBRIS. At this stage, melt begins to exit the vessel. The EJECT module that is currently being developed will provide the detailed treatment of the melt ejection and vessel blowdown process.

During the progression of the accident, especially after zircaloy oxidation causes rapid heatup of the core, the VICTORIA fission product model treats release of noble gases and vaporization of condensable fission products from the fuel

matrix. It also calculates vaporization of non-radioactive alloying species in the core. Then it calculates the behavior of these vapors and gases in each cell of the calculation--including aerosol formation, chemical reactions, and condensation on structures and pre-existing aerosols. It also acts in concert with FLUIDS to transport these materials throughout the system. VICTORIA has not yet been exercised within the integrated code. Nonetheless, we can describe some major expected stages in the release process.

While the clad remains on the fuel, it provides an effective barrier that diminishes fission product release. As a result, cesium, iodine and tellurium species released from the fuel must travel within the clad to failure locations to be released. During this process, much of the tellurium can be bound upon unoxidized inner clad surfaces as tin or zirconium tellurides depending on temperature.

As the heatup and core degradation processes continue, the cladding melts and potentially sparges some fission products from the fuel which can be carried along by the clad as it relocates. Thus, release from the clad can be important under some circumstances. The oxidized clad could also undergo local brittle failure. At that stage, the bare fuel stacks may crumble and form debris.

The debris slowly heats up and eventually melts. The large surface-to-volume ratio of the rubble and the coating of the rubble by melt as the melt relocates to cooler regions, coupled with the high debris temperatures, means that copious vapors are released--potentially including low volatility species such as barium, strontium, tin, indium, silver, etc. In the debris melting stage, release of fission products is limited by mass transport to the debris surface.

Following meltdown and slumping of the core, rapid fragmentation of hot melt may occur when it contacts water in the lower plenum. This is another potential source of rapid release of the more volatile species. After quenching of the melt, the debris release sequence repeats itself as the rubble heats up, melts and fails the vessel.

If failure occurs at high pressures, the blowdown process may entrain significant quantities of aerosols that had previously settled out on vessel surfaces. This yields a potentially large

"puff" release of fission products to the containment. In addition, the ejection process itself, finely aerosolizes a percentage of the melt, creating another source of fission products in the atmosphere promptly at failure.

Finally, volatile vapors which have condensed on surfaces (and on aerosols which have deposited on those surfaces) can reheat the surface locally to the point that they revaporize and are available for transport to containment.

All of the fission product release processes discussed above are treated by VICTORIA.

A couple of points need to be noted concerning this "typical" scenario. First, the code is not limited to the scenario described above. Rather, that scenario is fairly typical of the calculations performed to date, which include a number of variations of the station blackout accident (TMLB'). Second, as more scenarios are treated, we may expect significant deviations from the above scenario.

1.3 The Present Report

The purpose of this document is to describe the modules that have been developed for use in the MELPROG code and it is intended to serve as a reference for results that have already appeared in the literature and for future reports that will describe calculations that are currently under way. Due to the level of detail incorporated in each module, the MELPROG code can be used to extrapolate the current data base to a wide range of accident conditions. Since the MELPROG code is not limited to a particular accident sequence, but rather models all the important phenomena that can occur during a severe accident, it is ideally suited for accident management studies and analyses of the effects of specific plant features as well as "unusual" initial and boundary conditions on the course of accidents. For example, the MELPROG code is currently being used to investigate variations in the traditional station blackout accident (TMLB'): the effects of small breaks in the pump seals [1.12]; a stuck-open PORV; as well as early failure in the hot leg due to natural convection are all being considered.

Detailed discussion of the FLUIDS, CORE, STRUCTURES, RADIATION, DEBRIS, VICTORIA, IFCI and EJECT modules are provided in Sections 2 through 9, respectively. The FLUIDS and CORE

modules represent significant advances over the corresponding models employed in the one-dimensional MELPROG-PWR/MODO code [1.5]. In the current version of the code, the STRUCTURES, RADIATION and DEBRIS modules are very similar to the models in the MELPROG/MODO code. A new two-dimensional, mechanistic version of the DEBRIS meltdown model has been developed. The VICTORIA, IFCI and EJECT modules represent completely new development efforts. Currently, a one-way coupling exists between the VICTORIA module and CORE, STRUCTURES and FLUIDS: these modules provide VICTORIA with the information it needs to complete its calculation. In the future, the VICTORIA module will calculate the decay heat for the other modules. The IFCI, EJECT and two-dimensional DEBRIS modules are still in the developmental stage and will be included in future versions of the code. However, IFCI and the new DEBRIS module exist and are being used as stand-alone models. Further, they are currently being incorporated into developmental versions of MELPROG.

In addition, the MELPRO material properties routine is described in Section 10. The structure of the code and data storage techniques are discussed in Appendix A. Brief descriptions of the input deck and the graphics output are given in Appendices B and C, respectively. A detailed discussion of the input required to start a calculation and the output will be provided in a user's manual.

1.4 References

- 1.1 Reactor Safety Study--An Assessment of Accident Risks in US Commercial Nuclear Power Plants, WASH-1400, NUREG-75/014, U.S. Nuclear Regulatory Commission, Washington DC (1975).
- 1.2 J.A. Gieseke, et al., Source Term Code Package: A User Guide (MOD1), NUREG/CR-4587, Battelle Laboratories, Columbus, OH (July 1986).
- 1.3 R.M. Summers, "Degraded Core Modeling in MELCOR," (NUREG/CP-0080) Proc. Top. Mtg. on Reactor Physics and Safety, Saratoga Springs, NY, September 17-19 (1986).
- 1.4 C.M. Allison, et al., SCDAP/MOD1 Theory and Models, Draft Preliminary Report, Idaho National Engineering Laboratory, Idaho Falls, ID (January 1985).

- 1.5 W.J. Camp, et al., MELPROG-PWR/MODO: A Mechanistic Code for Analysis of Reactor Core Melt Progression and Vessel Attack Under Severe Accident Conditions, NUREG/CR-4909, SAND85-0237, Sandia National Laboratories, Albuquerque, NM (April 1987).
- 1.6 K.D. Bergeron, et al., User's Manual for CONTAIN 1.0, A Computer Code for Severe Nuclear Reactor Accident Containment Analysis, NUREG/CR-4085, SAND84-1204, Sandia National Laboratories, Albuquerque, NM (May 1985).
- 1.7 J.E. Kelly, et al., MELPROG/PWR-MOD1 Analysis of a TMLB' Accident Sequence, NUREG/CR-4742, SAND86-2175, Sandia National Laboratories, Albuquerque, NM (January 1987); "Integrated MELPROG/TRAC Analysis of PWR Station Blackout," R.J. Henninger, et al., AIChE Symposium Series, No. 257, Vol. 83, p. 326, (1987).
- 1.8 V.E. Denny, The CORMLT Code for Analysis of Degraded Core Accidents, EPRI NP-3767 CCM, Energy Power Research Institute, Palo Alto, CA (1984)
- 1.9 P.D. Bayless, Natural Circulation During A Severe Accident: Surry Station Blackout, EGG-SSRE-7858, Idaho National Engineering Laboratory, Idaho Falls, ID (1987).
- 1.10 R.O. Gauntt, et al., "Results of the ACRR-DFR Experiments," Proc. of the ANS/ENS Topical Meeting on Thermal Reactor Safety, San Diego, CA, February 2-6, 1986.
- 1.11 T.J. Heames and R.O. Gauntt, "MELPROG Analysis of the Debris Formation Experiment DF-1," Transactions of the Fifteenth Water Reactor Safety Meeting (NUREG/CP-0090), Gaithersburg, Maryland (October 1987).
- 1.12 W.J. Camp, et al., "Physical Modelling of the In-Vessel Phases of Severe Accidents--The MELPROG/TRAC Computer Code," International Symposium on Severe Accidents in Nuclear Power Plants, Sorrento, Italy (March 1988).

2.0 FLUIDS MODULE

The progression of an accident sequence in a commercial PWR through core meltdown involves the discrete motion of at least four different classes of materials. First, the motion of liquid water through the core determines the amount of heat removed and the rate of progression of the initial stages of the accident. Second, steam generated by boiling in the core tends to separate from the liquid, accumulating in the higher regions of the primary system. The relative distribution of steam and liquid water in the core is critical in determining core coolability. Third, oxidized and embrittled fuel pins can shatter, forming packed debris beds supported by the lower core support structure. And fourth, fuel pins, structure and debris beds can melt, redistributing material to the lower plenum. Gravity will always tend to separate these four material classes. A truly mechanistic model (one that does not rely on assumptions of gross phenomenology) must track the distribution, velocity and temperature of each of these material classes separately. This is done here by treating each of the four classes with continuum mechanics, and so is termed a "four-fluid" model.

The FLUIDS module solves the four fluid problem in a two-dimensional (r-z) representation of the reactor vessel, using information provided by the other modules in MELPROG. The solution for the first two fluids, steam and liquid water, can be coupled implicitly to the solution in a TRAC [2.1] representation of the ex-vessel reactor coolant system (RCS), providing an integrated treatment of the response of the entire plant to a given accident scenario. The FLUIDS module also solves a number of continuity equations in addition to those required for each of the four fluids. The corium mixture is divided into components (e.g., UO_2 , Zr, ZrO_2 , stainless steel, Ag-In-Cd, etc.) which are tracked separately, and the gaseous fission products provided by the link to VICTORIA are tracked within the reactor vessel and within the TRAC representation of the RCS.

The FLUIDS module provides a solution to the four-fluid problem on a fairly coarse-grid representation of the vessel. An accurate simulation of corium behavior during the complex processes of candling, debris bed formation, and energetic fuel coolant interactions require specific models for interfacial exchange terms and geometry degradation, applied on a relatively fine grid. The FLUIDS module provides the modules that specifically treat these phenomena with initial and boundary

conditions, as well as mass and energy exchange terms. In a sense it is the framework through which the other modules communicate with each other (some modules also communicate without going through FLUIDS), and with the rest of the RCS through the link to TRAC.

2.1 Basic Equations

The problem described above requires the simultaneous solution of the conservation equations of mass, axial momentum, radial momentum, and energy for each of the four fluids. The gas is modeled as a mixture of steam and hydrogen, so an additional hydrogen mass conservation equation is required. The four fluids are identified in the following equations by the index f , which is defined in Table 2.1.

Table 2.1 Fields Treated by Fluids	
<u>Fluid Number (f)</u>	<u>Material</u>
1	Gas Mixture of Steam and Hydrogen
2	Liquid Water
3	Solid Corium
4	Molten Corium

The differential forms of the equations will be listed and discussed. The finite difference forms will be given as part of the description of the solution algorithm that follows. The mass conservation equations have the following form for fluid f :

$$\frac{\partial}{\partial t}(\alpha_f \rho_f) + \nabla \cdot (\alpha_f \rho_f \vec{V}_f) - \sum_{k=1}^4 \Gamma_{kf} - \Gamma_f = 0 \quad (2.1)$$

α_f is the volume fraction of fluid f with respect to the total geometric cell volume (not flow volume). The non-flow volume fraction in the cell (made up of fuel pins and other vessel structural material) is designated α_s . This formulation is required because α_s can change with time as core structures deform and fall to lower regions of the vessel. The velocity vector $\vec{V}_f$ is composed of axial and radial components W_f and U_f , respectively. The third and fourth terms represent mass transfer among the fluids and external mass source terms, respectively. The mass transfer between steam and liquid water is treated implicitly in temperature and pressure, while the other mass transfers are treated explicitly and are provided by the other modules of MELPROG.

The hydrogen mass conservation equation is

$$\frac{\partial}{\partial t}(\alpha_1 \rho_h) + \nabla \cdot (\alpha_1 \rho_h \vec{V}_1) - \Gamma_h = 0 \quad . \quad (2.2)$$

The external mass source term Γ_h is produced by oxidizing zircaloy and stainless steel and is provided by the CORE, STRUCTURES and DEBRIS modules.

The axial momentum equations have the following form for fluid f:

$$\begin{aligned} \frac{\partial}{\partial t} W_f + \vec{V}_f \cdot \nabla W_f + \frac{1}{\rho_f} \frac{\partial P}{\partial z} \\ + \frac{1}{(\alpha \rho)_f} \sum_{k=1}^4 C_{kf} (W_f - W_k) |W_f - W_k| \\ + \frac{1}{(\alpha \rho)_f} D_f W_f |W_f| + g = 0 \quad . \end{aligned} \quad (2.3)$$

The fourth term represents momentum transfer among the fluids, while the fifth term represents wall friction. Coefficients C_{kf} and D_f are evaluated explicitly on the basis of the local flow regime. The radial momentum equations are completely analogous to Eq.(2.3), with the exception of gravitational acceleration

$$\begin{aligned} \frac{\partial}{\partial t} U_f + \vec{V}_f \cdot \nabla U_f + \frac{1}{\rho_f} \frac{\partial P}{\partial r} \\ + \frac{1}{(\alpha \rho)_f} \sum_{k=1}^4 C_{kf} (U_f - U_k) |U_f - U_k| \\ + \frac{1}{(\alpha \rho)_f} D_f U_f |U_f| = 0 \quad . \end{aligned} \quad (2.4)$$

The energy equations are given by

$$\begin{aligned} \frac{\partial}{\partial t}(\alpha_f \rho_f u_f) + \nabla \cdot (\alpha_f \rho_f u_f \vec{V}_f) + P \left(\frac{\partial \alpha_f}{\partial t} + \nabla \cdot \alpha_f \vec{V}_f \right) \\ - \sum_{k=1}^4 \Gamma_{kf} h_{sf} - \sum_{k=1}^4 Q_{kf} - Q_f = 0 \quad . \end{aligned} \quad (2.5)$$

The third term is the work term. The fourth term represents energy exchange between the fluids due to phase change with h_{sf} representing the saturation enthalpy of fluid f . The fifth term represents heat transfer between fluids. The sixth term represents external energy sources provided by other modules of MELPROG.

Finally, a constraint on the sum of the fluid volume fractions is required:

$$1 - \sum_{f=1}^4 \alpha_f - \alpha_s = 0 \quad . \quad (2.6)$$

Equations (2.1) through (2.6) are a set of seventeen coupled, non-linear, partial differential equations that, along with material equations of state and relations for mass, energy and momentum exchange, provide the basis for a solution of the problem. The description of the solution algorithm that follows assumes that all four fluids are in fact present throughout the reactor vessel. Fortunately this is rarely true, so that a significant reduction in the complexity of the problem is usually possible. The steps taken to maximize this reduction in complexity will be described later.

2.2 Solution Algorithm

The algorithm used to solve this equation set is based on the Stability Enhancing Two-Step (SETS) [2.2] method, which is used to solve the one-dimensional, two-fluid problem in TRAC/PF1. SETS has enhanced numerical stability characteristics that allow the material Courant condition to be exceeded under some conditions, resulting in longer computational time steps and a less expensive calculation. Compared to the usual semi-implicit method (for example, as used in the TRAC/PF1 three-dimensional vessel component) [2.1], SETS also has enhanced stability characteristics for problems with phase change, because of a more implicit treatment of momentum coupling between the fluids and of the partition of volume fraction within a cell. The algorithm was developed partly from the previous one-dimensional, three-fluid model in the MIMAS code [2.3], which also utilized SETS. The algorithm consists of three main parts: the momentum equation solver, the basic step, and the mass and energy stabilizers. These parts are described in detail, followed by a discussion of the treatment of low volume fractions.

2.2.1 Momentum Equations

In the fluid dynamics calculation, a staggered grid [2.4] is used for the spatial finite-differencing of the differential equations. In this scheme, volume fractions and fluid state variables are evaluated using cell-centered (i,j) control volumes, while velocities are evaluated using cell-edge centered control volumes. Thus all the terms in the axial momentum Eq.(2.3) are evaluated using control volumes centered at $(i,j+1/2)$ and all the terms in the radial momentum Eq.(2.4) using control volumes centered at $(i+1/2,j)$.

Terms in Eq.(2.3) requiring further finite difference expansion are the following:

$$\left(\frac{1}{\rho_f} \frac{\partial P}{\partial z}\right)_{i,j+1/2} = \frac{1}{1/2(\rho_{i,j} + \rho_{i,j+1})_f} \frac{(P_{i,j+1} - P_{i,j})}{1/2(\Delta z_j + \Delta z_{j+1})} , \quad (2.7)$$

$$(\vec{V} \cdot \nabla W)_{i,j+1/2} = \frac{W_{i,j+1/2} \langle W \rangle_z}{1/2(\Delta z_j + \Delta z_{j+1})} + \frac{U_{i,j+1/2} \langle W \rangle_r}{1/2(\Delta r_i + \Delta r_{i+1})} , \quad (2.8)$$

where

$$\begin{aligned} \langle W \rangle_z &= W_{i,j+1/2} - W_{i,j-1/2} \quad \text{if} \quad W_{i,j+1/2} \geq 0 , \\ \langle W \rangle_z &= W_{i,j+3/2} - W_{i,j+1/2} \quad \text{if} \quad W_{i,j+1/2} < 0 ; \end{aligned} \quad (2.9)$$

$$\begin{aligned} \langle W \rangle_r &= W_{i,j+1/2} - W_{i-1,j+1/2} \quad \text{if} \quad U_{i,j+1/2} \geq 0 , \\ \langle W \rangle_r &= W_{i+1,j+1/2} - W_{i,j+1/2} \quad \text{if} \quad U_{i,j+1/2} < 0 ; \end{aligned} \quad (2.10)$$

and

$$\frac{1}{(\alpha\rho)_{i,j+1/2}} = \frac{2}{[(\alpha\rho)_{ij} + (\alpha\rho)_{i,j+1}]} . \quad (2.11)$$

Note that W is calculated at the staggered grid points $(i,j+1/2)$, whereas U is calculated at $(i+1/2,j)$. In order to determine $U_{i,j+1/2}$ in Eqs.(2.8,2.10), U is averaged over the four neighboring nodes as follows:

$$U_{i,j+1/2} = \frac{(U_{i+1/2,j} + U_{i+1/2,j+1} + U_{i-1/2,j} + U_{i-1/2,j+1})}{4} . \quad (2.12)$$

Analogous expressions are used for terms in the radial momentum Eq.(2.4). These spatial finite difference forms will not be repeated so that the following equations can clearly show the different temporal differencing and linearization schemes.

The goal in this step is to reduce Eq.(2.3) and (2.4) to simple linear relationships between each local fluid velocity and an associated pressure difference. This can be done easily by evaluating all the terms except the nonsteady term using old-time level velocities, but this fully explicit technique has limited stability characteristics. A fully implicit, iterative solution is also possible, but very expensive. SETS provides an intermediate method, in which successive linearizations provide separate implicit solutions for the major couplings--among the fluid velocities in each cell and convectively between cells for each fluid.

The solution procedure for the axial momentum equations will be discussed, with an analogous procedure used for the radial momentum equations. In the first linearization old-time level velocities are used in the convective terms, decoupling the equations spatially.

$$\begin{aligned} & \frac{\bar{w}_f^{n+1} - w_f^n}{\Delta t} + \vec{v}_f^n \cdot \nabla \bar{w}_f^n + \frac{1}{\rho_f^n} \left(\frac{\partial P}{\partial z} \right)^n \\ & + \frac{1}{(\alpha \rho)_f^n} \sum_{k=1}^4 C_{kf} \left\{ 2(\bar{w}_f^{n+1} - \bar{w}_k^{n+1}) - (w_f^n - w_k^n) \right\} |w_f^n - w_k^n| \\ & + \frac{1}{(\alpha \rho)_f^n} D_f (2\bar{w}_f^{n+1} - w_f^n) |w_f^n| + g = 0 \end{aligned} \quad (2.13)$$

where $\bar{w}$ is a predictor velocity and w is a stabilizer velocity. Equation (2.13) represents a 4x4 system of equations in $\bar{w}^{n+1}$ at each mesh cell, which are solved directly. These predictor velocities represent the effect of coupling between the fluids within each cell. The third and fourth terms in Eq.(2.13) represent temporal expansions of the analogous terms in Eq.(2.3), --see Ref. [2.2].

The second linearization uses the predictor velocities for the intracell fluid interaction terms (decoupling the fluid equations from each other), while solving for the convective terms.

$$\begin{aligned}
& \frac{\tilde{W}_f^{n+1} - W_f^n}{\Delta t} + \vec{V}_f^n \cdot \nabla \tilde{W}_f^{n+1} + \frac{1}{\rho_f^n} \left(\frac{\partial P}{\partial z} \right)^n \\
& + \frac{1}{(\alpha \rho)_f^n} \sum_{k=1}^4 C_{kf} \left\{ 2 \left(\tilde{W}_f^{n+1} - \tilde{W}_k^{n+1} \right) - \left(W_f^n - W_k^n \right) \right\} \left| W_f^n - W_k^n \right| \\
& + \frac{1}{(\alpha \rho)_f^n} D_f \left(2 \tilde{W}_f^{n+1} - W_f^n \right) \left| W_f^n \right| + g = 0 \quad . \quad (2.14)
\end{aligned}$$

Equation (2.14) represents four uncoupled systems of equations in $\tilde{W}^{n+1}$, which are solved using the methods discussed in the next section. These velocities are called stabilizer velocities, and represent the effects of coupling between cells.

The third linearization uses stabilizer velocities for the convective terms, with new time-level pressures:

$$\begin{aligned}
& \frac{W_f^{n+1} - W_f^n}{\Delta t} + \vec{V}_f^n \cdot \nabla \tilde{W}_f^{n+1} + \frac{1}{\rho_f^n} \left(\frac{\partial P}{\partial z} \right)^{n+1} \\
& + \frac{1}{(\alpha \rho)_f^n} \sum_{k=1}^4 C_{kf} \left\{ 2 \left(W_f^{n+1} - W_k^{n+1} \right) - \left(W_f^n - W_k^n \right) \right\} \left| W_f^n - W_k^n \right| \\
& + \frac{1}{(\alpha \rho)_f^n} D_f \left(2 W_f^{n+1} - W_f^n \right) \left| W_f^n \right| + g = 0 \quad . \quad (2.15)
\end{aligned}$$

Equation (2.15) also represents a 4x4 system in velocities at each mesh cell, but these systems are solved with the new time level pressures factored out on the right-hand sides to give equations of the form:

$$W_{f,i,j+1/2}^{n+1} = a_{f,i,j+1/2} + b_{f,i,j+1/2} \left(P_{i,j+1}^{n+1} - P_{i,j}^{n+1} \right) \quad . \quad (2.16)$$

Equation (2.16) is the required relationship between new time-level pressures and velocities. The linear coefficients a and b are held constant during the next step. An analogous procedure for the radial momentum equations yields a similar equation set for the radial velocities:

$$U_{f,i+1/2,j}^{n+1} = a_{f,i+1/2,j} + b_{f,i+1/2,j} \left(P_{i+1,j}^{n+1} - P_{i,j}^{n+1} \right) \quad . \quad (2.17)$$

2.2.2 Basic Step

All terms in the mass and energy Eqs.(2.1), (2.2) and (2.5) are evaluated using control volumes at cell centers (i,j). Terms in these equations requiring further finite difference expansion follow:

$$\frac{\partial}{\partial t}(\alpha_f \rho_f) = \frac{\alpha_f^{n+1} \rho_f^{n+1} - \tilde{\alpha}_f^n \tilde{\rho}_f^n}{\Delta t} , \quad (2.18)$$

where $\tilde{\alpha}^n$ and $\tilde{\rho}^n$ are calculated in the mass stabilizer step of the previous time step;

$$\frac{\partial}{\partial t}(\alpha_f \rho_f u_f) = \frac{\alpha_f^{n+1} \rho_f^{n+1} u_f^{n+1} - \tilde{\alpha}_f^n \tilde{\rho}_f^n \tilde{u}_f^n}{\Delta t} , \quad (2.19)$$

where $\tilde{u}^n$ is calculated in the energy stabilizer step of the previous time step.

The second term in Eqs.(2.1), (2.2), and (2.5) has the form $\nabla \cdot (YV)$ and is evaluated as:

$$\begin{aligned} \nabla \cdot (YV) = & \left[F_{i,j+1/2} \langle Y, W \rangle_{i,j+1/2} - F_{i,j-1/2} \langle Y, W \rangle_{i,j-1/2} \right. \\ & \left. + F_{i+1/2,j} \langle Y, U \rangle_{i+1/2,j} - F_{i-1/2,j} \langle Y, U \rangle_{i+1/2,j} \right] / VOL_{i,j} \end{aligned} \quad (2.20)$$

where F represents flow areas in the axial direction when the subscript is of the form i+1/2,j and in the radial direction when the subscript is of the form i,j+1/2, and where

$$\begin{aligned} \langle Y, W \rangle_{i,j+1/2} &= \frac{Y_{i,j}}{(1-\alpha_s)_{i,j}} W_{i,j+1/2} \quad \text{if } W_{i,j+1/2} \geq 0 \\ &= \frac{Y_{i,j+1}}{(1-\alpha_s)_{i,j+1}} W_{i,j+1/2} \quad \text{if } W_{i,j+1/2} < 0 . \end{aligned} \quad (2.21)$$

Similar upwind difference forms hold for the other terms in Eq.(2.20).

For the mass conservation Eqs.(2.1) and (2.2), $Y_{i,j}$ is given by $\alpha^{n+1} \rho^{n+1}$. The neighboring cell Y's are given by stabilized mass densities, which are calculated in the mass stabilizer step of the previous time step. Similarly, for the energy conser-

vation Eq.(2.5), $Y_{i,j}$ is given by $\alpha^{n+1} \rho^{n+1} u^{n+1}$, but the neighboring cell Y's are given by stabilized energy densities, which are calculated in the energy stabilizer step of the previous time step. Using different time level values in the convection terms is nonconservative, but conservation is maintained by the mass and energy stabilizer steps. Using new time-level values of $\alpha \rho$ and $\alpha \rho u$ in the convection terms increases stability, while using old-time level values for the neighboring cells reduces the spatial coupling of the system of equations that must be solved (they are still coupled by pressures).

The work term in Eq.(2.5) is given by

$$p^{n+1} \left[\frac{\alpha_f^{n+1} - \alpha_f^n}{\Delta t} + \nabla \cdot \left(\alpha_f^n \vec{V}_f^{n+1} \right) \right]$$

where the convection term is upwind-differenced as in Eq.(2.20).

Equations (2.16) and (2.17) are used to eliminate velocities in Eqs.(2.1), (2.2), and (2.5). Equations (2.1), (2.2), (2.5), and (2.6) now constitute a set of ten equations for each mesh cell. With material equations of state used to give internal energy and density in terms of pressure and temperature, these ten equations are functions of the pressure field (cell and neighboring cell pressures), cell partial pressure of hydrogen, and cell values of volume fraction and temperatures for each of the four fluids.

These systems are solved by Newton iteration using the following procedure (which was developed for the two-fluid systems of TRAC/PF1). Variations in the ten equations with respect to cell pressure are divided into those caused by density changes and those caused by velocity changes. The Jacobian matrix is partitioned so that variations with respect to cell variables are included on the left-hand side of Eq.(2.22), while variations with respect to neighboring cell pressures and the terms for variation of velocity with respect to cell pressure are included on the right hand side.

$$\begin{aligned} \underline{A} \delta \vec{x} = & -\vec{B}_1 \delta P_{i,j} - \vec{B}_2 \delta P_{i+1,j} - \vec{B}_3 \delta P_{i-1,j} \\ & - \vec{B}_4 \delta P_{i,j+1} - \vec{B}_5 \delta P_{i,j-1} - \vec{S} \quad , \end{aligned} \quad (2.22)$$

where $\underline{A}$ is the 10 x 10 Jacobian matrix

$$\underline{A} = \begin{pmatrix} \frac{\partial M_1}{\partial P} & \frac{\partial M_1}{\partial \alpha_1} & \cdots & \frac{\partial M_1}{\partial \alpha_4} & \frac{\partial M_1}{\partial T_1} & \cdots & \frac{\partial M_1}{\partial T_4} & \frac{\partial M_1}{\partial P_h} \\ \vdots & & & & & & & \vdots \\ \frac{\partial M_5}{\partial P} & & & & & & & \vdots \\ \frac{\partial E_1}{\partial P} & & & & & & & \\ \vdots & & & & & & & \\ \frac{\partial E_4}{\partial P} & & & & & & & \vdots \\ \frac{\partial AL}{\partial P} & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \frac{\partial AL}{\partial P_h} \end{pmatrix} \quad (2.23)$$

where the M_i represents Eq.(2.1) ($i = 1$ to 4) and Eq.(2.2) ($i = 5$). The E_i represent Eq.(2.5) ($i = 1$ to 4), AL represents Eq.(2.6), and

$$\vec{\delta x} = (\delta P, \delta \alpha_1, \dots, \delta \alpha_4, \delta T_1, \dots, \delta T_4, \delta P_h) \quad , \quad (2.24)$$

$$\vec{B}_1 = \left(\frac{\partial M_1}{\partial P_{i,j}}, \dots, \frac{\partial M_5}{\partial P_{i,j}}, \frac{\partial E_1}{\partial P_{i,j}}, \dots, \frac{\partial E_4}{\partial P_{i,j}}, \frac{\partial AL}{\partial P_{i,j}} \right) \quad , \quad (2.25)$$

$$\vec{S} = (M_1, \dots, M_5, \dots, E_1, \dots, E_5, \dots, AL) \quad , \quad (2.26)$$

and $\underline{A}$ and $\vec{S}$ are evaluated using previous iterate values for the 10 independent variables, and $\vec{S}$ represents the residuals of the 10 equations. Forms similar to Eq.(2.25) exist for the other B_i in Eq.(2.22). Equation (2.22) is solved for $\vec{\delta x}$ directly.

$$\begin{aligned} \vec{\delta x} = & -\underline{A}^{-1} \vec{B}_1 \delta P_{i,j} - \underline{A}^{-1} \vec{B}_2 \delta P_{i+1,j} - \underline{A}^{-1} \vec{B}_3 \delta P_{i,j+1} \\ & - \underline{A}^{-1} \vec{B}_4 \delta P_{i,j-1} - \underline{A}^{-1} \vec{S} \quad , \end{aligned} \quad (2.27)$$

The first row of Eq.(2.27) is an implicit relationship between cell pressure variations and neighboring cell pressure variations. When the first rows of Eq.(2.27) are collected for all the mesh cells, an implicit system of equations for corrections to the pressure field results, which is solved by using the methods in the next section. These pressure corrections are then substituted back into Eq.(2.27) for each mesh cell to give the rest of δx . These corrections are then

applied to pressures, hydrogen partial pressures, and the four-fluid volume fractions and temperatures. Equations (2.16) and (2.17) are used to update velocities. If the maximum pressure correction is greater than a specified convergence criterion, another Newton iteration is performed.

2.2.3 Mass and Energy Stabilizers

Equations (2.1), (2.2) and (2.5) are converted to the following forms, respectively:

$$\frac{\tilde{\alpha}_f^{n+1}\tilde{\rho}_f^{n+1}-\tilde{\alpha}_f^n\tilde{\rho}_f^n}{\Delta t} + \nabla \cdot \left(\tilde{\alpha}_f^{n+1}\tilde{\rho}_f^{n+1}\vec{V}_f^{n+1} \right) - \sum_{k=1}^4 \Gamma_{kf} - \Gamma_f = 0 \quad , \quad (2.28)$$

$$\frac{\tilde{\alpha}_h^{n+1}\tilde{\rho}_h^{n+1}-\tilde{\alpha}_h^n\tilde{\rho}_h^n}{\Delta t} + \nabla \cdot \left(\tilde{\alpha}_h^{n+1}\tilde{\rho}_h^{n+1}\vec{V}_h^{n+1} \right) - \Gamma = 0 \quad , \text{ and} \quad (2.29)$$

$$\begin{aligned} & \frac{\tilde{\alpha}_f^{n+1}\tilde{\rho}_f^{n+1}\tilde{u}_f^{n+1}-\tilde{\alpha}_f^n\tilde{\rho}_f^n\tilde{u}_f^n}{\Delta t} + \nabla \cdot \left(\tilde{\alpha}_f^{n+1}\tilde{\rho}_f^{n+1}\tilde{u}_f^{n+1}\vec{V}_f \right) \\ & + P^{n+1} \left(\frac{\tilde{\alpha}_f^{n+1}-\tilde{\alpha}_f^n}{\Delta t} + \nabla \cdot \tilde{\alpha}_f^n \vec{V}_f^{n+1} \right) - \sum_{k=1}^4 \Gamma_{kf} h_{sf} - \sum_{k=1}^4 Q_{kf} - Q_f = 0 \quad , \end{aligned} \quad (2.30)$$

where all parameters are set using the new values determined in the basic step. These equations are solved for stabilized values of $(\alpha\rho)^{n+1}$ and $(\alpha\rho u)^{n+1}$ for each fluid and for hydrogen, using the methods described in the next section. These parameters provide stabilized mass and energy densities for the neighboring cell convective terms in the next time step.

2.3 Imbedded Paths

Modeling the vessel in cylindrical coordinates is justified because most of the structural geometry is approximately radially symmetric, the main exceptions being the locations of the hot and cold leg nozzles. Some structural elements, however, create a flow geometry that cannot be modeled in two dimensions. Material can flow inside the control rod guide tube covers between the upper head and upper plenum without interacting with the two-dimensional flow field on the outside of the control rod covers. This internal flow could possibly have a significant effect on the partition of energy between the upper head and upper plenum.

To allow this additional mass and energy exchange to occur in the model, the capability of additional, user-defined interfaces between cells have been added.

This capability allows the magnitude of mixing due to natural convection between the upper head and upper plenum to be calculated, but not the details of the flow within the control rod guide tubes. The only additions to the two-dimensional grid are new interfaces between nonadjoining cells, no new fluid volumes are added. A more detailed treatment would require additional fluid volumes that interact thermally with the two-dimensional grid; this may be done in the future using the TRAC link.

2.4 Methods for Solving Systems of Equations

The axial velocity stabilizer Eq.(2.14), the equivalent radial stabilizer velocity equations, the pressure correction equation constructed by collecting the first rows of Eq.(2.27), and the mass and energy stabilizer Eqs.(2.28) and (2.29) are all linear systems of equations of the form $\underline{A} \vec{x} = \vec{b}$. The dimension of the matrix $\underline{A}$ is the number of cells in the vessel, typically fifty to one hundred. Because of the relatively large size of these systems, and the number of times they must be solved per time step (depends on the number of fluids present in the vessel and the number of iterations of the basic step), considerable effort has been extended in making the solution as efficient as possible.

Iterative methods have been tried with some success, but their convergence cannot be guaranteed under all conditions, particularly when the time step is large or the problem is separated into completely isolated and independent regions. Currently two types of direct solution are implemented in the code. The first is an LUD decomposition of the full matrix $\underline{A}$, then back substitution using the right-hand side vector $\vec{b}$. The second utilizes the fact that the matrix $\underline{A}$ is nearly banded. The vessel is modeled in two dimensions, so in general $\underline{A}$ is a five-stripe banded matrix, with a limited number of non-zero elements off the bands. The "outliers" are generated by connecting non-adjacent axial levels. This occurs when imbedded paths are used to simulate flow within the control rod guide tubes (next section), or when TRAC junctions create a flow path between non-adjacent levels. The matrix $\underline{A}$ can be partitioned into the banded matrix and a matrix composed of the outliers, with the solution

for the banded system modified by straight-forward linear algebra. This method is very efficient when the number of outliers is small (less than about one-fourth of the dimension of A), as is usually true in a typical plant model.

2.5 Logic for Equation Set Switching

The numerical solution of a multi-field problem requires great care as the volume fraction of one of the fluids approaches zero in a cell. For many reasons, not the least of which is round-off error, it is necessary to define a volume fraction below which the fluid is said to no longer exist. This cut-off volume fraction is set on input, with a typical value of 10^{-5} .

The logic that determines whether or not a fluid is present in a cell makes use of the mass stabilizer Eqs.(2.28) and (2.29). If the solution of these equations shows that, because of convection or source terms, a fluid has just become present in a cell, then the appropriate flag (or "knob") is set. These flags can also be set on the basis of a comparison of fluid temperature and saturation temperature. When the appropriate flag is set, the mass and energy equations for that fluid (or for hydrogen) are added to the system of Eq.(2.22) for that cell. The continuity equations solved in the basic step in Eq.(2.22) cannot be used to predict the existence of a fluid in a cell, because they are not solved when the fluid does not exist. The number of equations represented by Eq.(2.22) can therefore range from 2 (single fluid) to 10 (four fluids, hydrogen, and volume fraction constraint). It is very often the case in a typical accident calculation that only one fluid exists in a cell, so the capability of reducing the dimension of the Jacobian matrix in Eq.(2.22) results in a great increase in computing efficiency.

Similar flags are set for momentum equations. The spatially decoupled Eqs.(2.13) and (2.15) are adjusted if a fluid does not exist in either cell bordering the cell interface. The adjustments produce realistic results when the fluid does become present. These flags can also be set to indicate the presence of a wall.

Global flags are set if a fluid is present anywhere in the vessel. The momentum stabilizer Eq.(2.14) and the energy stabilizer Eq.(2.30) are not solved if these flags are not set.

2.6 Time Step Control

The following logic is used to determine the time step used during a MELPROG calculation. If MELPROG is coupled to TRAC, then this time step is sent to TRAC. If the MELPROG-determined time step is less than the TRAC-determined time step, then the MELPROG time step will be used for the linked calculation.

The time step is increased by a small factor (typically 1.05 to 1.10, set on input) over the previous time step, unless limited by one of the possible constraints listed below:

1. The time step can be limited by a maximum time step, set on input as a function of problem time. The user provides this input in order to avoid the possible non-physical results of an excessive time step.
2. The time step can be limited by a maximum Courant number, set on input. Presently the Courant number is limited to unity when VICTORIA is run in MELPROG. In general, calculations with a Courant number greater than ten may not be sufficiently accurate.
3. The time step can be limited by maximum rates of temperature and volume fraction change, set on input. These constraints can either prevent the time step from increasing, or cut it by a factor of two, depending on the rate of change.
4. The time step can be limited by rates of energy and mass transfer that are explicitly determined by other modules.
5. The time step will not increase if the number of iterations of the basic step exceeds an input criterion (typically four). The time step will be decreased by a factor of two if convergence is not achieved in the basic step, either because of too many iterations (typically ten), or because a negative volume fraction has been calculated.

This logic has been developed, largely by trial and error, in order to maximize the efficiency of a calculation. A common mode of failure of a calculation is reduction of the time step below a minimum set on input to a very small number. This can indicate a coding error, or that the problem should be run with a smaller time step. The above logic is, unfortunately, not

completely general, and there are times when the user must determine the maximum usable time step. Even this time step is often not optimum, because the code may run with less work per step with a smaller time step.

2.7 Equation of State

The changes to the MELPROG equation-of-state package for water consisted of (1) installing the latest version of the equation-of-state package (EOS) from the TRAC systems code, which has been improved over the original version used in the MELPROG package, and (2) adding analytic extensions at the limits of the TRAC package. The EOS package, which is collectively referred to in MELPROG and TRAC as THERMO, consists of a series of analytic fits to the steam tables. In normal operation, THERMO returns water and steam properties, plus property derivatives with respect to the independent variables, which are the water and vapor temperatures and the total and partial steam pressures. When the input variables exceed the limits of the regions over which the analytic fits are valid, THERMO returns the properties and derivatives at the limits. Unfortunately, the pressure iteration step in the SETS method now is receiving nonzero derivatives for properties that are actually not changing; as a consequence, the pressure iteration will either fail or reduce the time step drastically. Adding the analytic extensions to the THERMO package allows the properties to continue to change in agreement with the derivatives. The extensions were added because of the likelihood of generating supercritical conditions during an FCI. They are also useful when supercritical temperatures occur in the water field, which is a common occurrence during calculation of severe accident sequences with MELPROG.

The limits in the current TRAC version of the THERMO package are shown in Table 2.2.

Table 2.2 TRAC Water Equation of State Limits	
Minimum Hydrogen Partial Pressure	1×10^{-5} Pa
Minimum Steam Partial Pressure	1 Pa
Maximum Steam Partial Pressure	45 MPa
Minimum Vapor Temperature	273 K
Maximum Vapor Temperature	3000 K
Minimum Water Temperature	273 K
Maximum Water Temperature	713.9 K
Maximum Saturation Temperature	647 K

These limits are extended as follows:

1. If either the vapor temperature or pressure is off the table limits, then the vapor equation of state is extended assuming ideal gas behavior. The equation used is

$$\rho_s^i = \frac{P_s}{R_s T_v} , \quad (2.31)$$

where

$$\begin{aligned} \rho_s^i &= \text{ideal gas law steam density, kg/m}^3, \\ P_s &= \text{steam partial pressure, Pa,} \\ R_s &= \text{gas constant for steam} = 462 \text{ J/kg/K, and} \\ T_v &= \text{vapor temperature, K.} \end{aligned}$$

This equation is joined smoothly to the table value at the limit by adding an offset: $\rho_s = \rho_s^i + \Delta\rho$. When the pressure is off the table, $\Delta\rho$ is given by:

$$\Delta\rho = \rho_s^\circ(P_s^\circ, T_v) - \frac{P_s^\circ}{R_s T_v} , \quad (2.32a)$$

where the superscript "°" refers to the table edge values; and when the vapor temperature is off the table, it is given by:

$$\Delta\rho = \rho_s^\circ(P_s, T_v^\circ) - \frac{P_s}{R_s T_v^\circ} . \quad (2.32b)$$

The derivatives of vapor density with respect to temperature and pressure are replaced by the ideal gas law derivatives if either of the independent variables (T_v and P_s) is outside the table limits. When the pressure is off the table, the pressure derivative is given by:

$$\frac{\partial \rho_s}{\partial P_s} = \frac{1}{R_s T_v} ; \quad (2.33a)$$

and when the temperature is off the table, it is given by:

$$\frac{\partial \rho_s}{\partial P_s} = \frac{1}{R_s T_v} - \frac{1}{R_s T_v^\circ} + \left(\frac{\partial \rho_s}{\partial P_s}(T_v^\circ, P_s) \right) . \quad (2.33b)$$

When the temperature is off the table, the temperature derivative is given by:

$$\frac{\partial \rho_s}{\partial T_v} = - \frac{P_s}{R_s T_v^2} ; \quad (2.34a)$$

and when the pressure is off the table, it is given by:

$$\frac{\partial \rho_s}{\partial T_v} = - \frac{P_s}{R_s T_v^2} + \frac{P_s^\circ}{R_s T_v^2} + \left(\frac{\partial \rho_s}{\partial T_v} (T_v, P_s^\circ) \right) . \quad (2.34b)$$

The internal energy is extended using first order Taylor series expansions in T and P:

$$u_s = u_s^\circ + \left(\frac{\partial u_s}{\partial T_v} \right)^\circ \Delta T_v + \left(\frac{\partial u_s}{\partial P_s} \right)^\circ \Delta P_s ; \Delta P_s = P_s - P_s^\circ ; \Delta T_v = T_v - T_v^\circ . \quad (2.35)$$

2. If either the liquid temperature or the pressure are off the table limits, then the liquid internal energy and density are extended using Taylor series in T and P:

$$u_l = u_l^\circ + \left(\frac{\partial u_l}{\partial T_l} \right)^\circ \Delta T_l + \left(\frac{\partial u_l}{\partial P_l} \right)^\circ \Delta P_l ; \quad (2.36)$$

and

$$\rho_l = \rho_l^\circ + \left(\frac{\partial \rho_l}{\partial T_l} \right)^\circ \Delta T_l + \left(\frac{\partial \rho_l}{\partial P_l} \right)^\circ \Delta P_l ; \quad (2.37)$$

where $\Delta T_l = T_l - T_l^\circ$ and $\Delta P_l = P_l - P_l^\circ$. The enthalpies of water and steam at saturation and the saturation temperature must also be extended in pressure, as the derivatives of these quantities, as calculated by THERMO, do not quite go to zero in the table at the critical point. In the extension, they are allowed to keep changing slowly with pressure and multiplied by a function that gradually decreases the change in the properties and the property derivatives to zero. The actual expressions for the enthalpies at saturation are

$$h_{sv} = h_{sv}^\circ f_h , \quad h_{sl} = h_{sl}^\circ f_h , \quad f_h = 1 + 10^6 \left(\frac{1}{P_s^\circ} - \frac{1}{P_s} \right) , \quad (2.38)$$

and the derivatives are given as

$$\frac{\partial h_{sv}}{\partial P} = \frac{\partial h_{sv}^o}{\partial P} f_h - h_{sv} \frac{10^6}{p_s^2}, \quad \frac{\partial h_{sl}}{\partial P} = \frac{\partial h_{sl}^o}{\partial P} f_h - h_{sl} \frac{10^6}{p_s^2}. \quad (2.39)$$

Some modification of the TRAC liquid density routine was required, as the fit used for the liquid density had a positive derivative with respect to temperature near the critical point; this is completely nonphysical, and caused the heat transfer routines to calculate a negative Grashof number. All of the transport properties are restricted in MELPROG to pressures less than the critical pressure to avoid erroneous results. It was also found that the liquid heat capacity and thermal conductivity would go negative at enthalpies greater than that at the critical point; in MELPROG, the liquid enthalpy passed to the liquid heat capacity and thermal conductivity routines (CPLL and THCL, respectively), is restricted to a maximum of 2.099×10^6 J/kg.

2.8 Auxiliary Equations Required for Closure

A number of auxiliary or constitutive relations must be specified in order to solve Eqs.(2.1-2.6). In addition to an equation of state for each field, correlations for the wall shear, interfacial heat transfer, interfacial shear and wall heat transfer must be prescribed. These correlations are used to calculate C_{kf} and D_f in Eqs.(2.3,2.4) as well as Q_{kf} in Eq.(2.5).

Correlations for drag, heat transfer as well as the interfacial contact area between the vapor and liquid water fields are chosen on the basis of flow regime maps given in the TRAC code [2.1]. Brief discussions of these maps are provided in Section 2.8.1: the specific correlations that are used in each regime are given in Appendix D (Section D.2).

Section 2.8.2 discusses wall shear and wall heat transfer correlations for the water and vapor fields. The current version of the FLUIDS module neglects heat transfer and drag between the walls and the corium fields. However, in some cases, these effects are considered in other MELPROG modules. For example, the CORE module treats heat transfer as well as drag between core structures and molten materials (as these materials candle downward). Also, whenever corium (or debris) is stationary and dry, the DEBRIS module considers heat transfer between the debris and adjacent structures.

Interfacial heat transfer correlations for the corium fields are discussed in Section 2.8.3. Drag forces between the corium and vapor field are modeled in the manner suggested by Lipinski [2.5].

2.8.1 Flow Regime Maps

The vertical flow map (see Fig. 2.1) is a simplified version of the one used in TRAC [2.1]. This simplified map does not depend on mass flux; only the normalized vapor volume fraction, α , is used to determine when the flow regime changes. This normalized volume fraction is defined as

$$\alpha = \frac{\alpha_v}{\alpha_v + \alpha_l} \quad (2.40)$$

As seen in Fig. 2.1, the flow regime is bubbly flow when the volume fraction is below 0.3. From 0.3 to 0.5, the flow regime is slug flow. Above 0.75, the regime is annular mist flow. The region from 0.5 to 0.75 is a transition regime between slug and annular mist flow and the heat transfer coefficients are determined by a smooth interpolation scheme based on the values in the neighboring regimes.

In most applications of interest, vertical flow is dominant. However, a horizontal flow map is sometimes needed (especially when the liquid level in a cell must be determined). A flow regime map based on work done by Baker [2.6] is shown in Fig. 2.2. This map is based on corrected vapor and liquid mass fluxes, G'_v and G'_l , respectively, that are given by:

$$G'_v = \frac{G_v}{\lambda} \quad , \quad G'_l = G_l \psi \quad (2.41)$$

where the factors λ and ψ are used to adjust for different liquid and vapor properties, as the original map is for air-water flows. These factors are defined as [2.6]:

$$\lambda = \left[\frac{\rho_v}{\rho_A} \frac{\rho_L}{\rho_W} \right]^{\frac{1}{2}} \quad , \quad \psi = \left[\frac{\sigma_W}{\sigma} \right] \left[\frac{\mu_L}{\mu_W} \left[\frac{\rho_W}{\rho_L} \right]^2 \right]^{\frac{1}{3}} \quad (2.42)$$

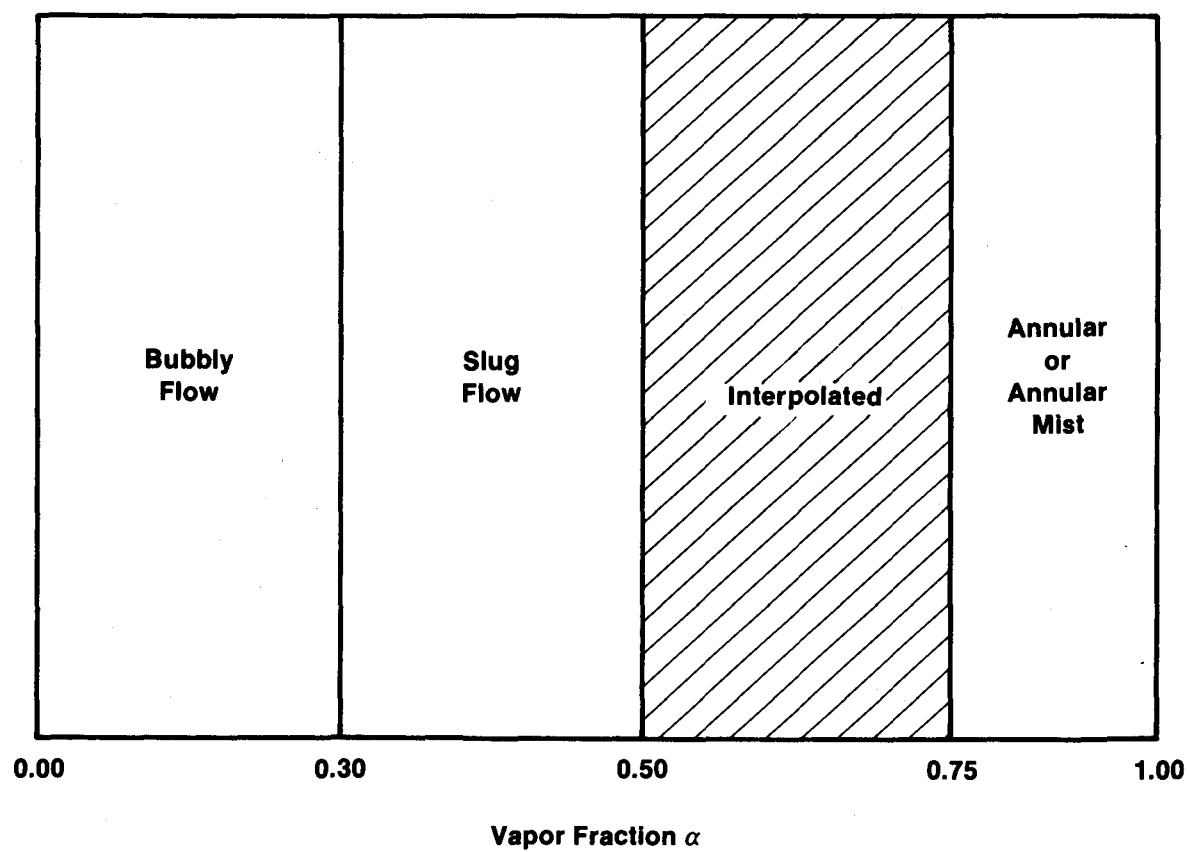


Fig. 2.1 Flow-Regime Map for Two-Phase Vertical Flow.

where

- ρ_x = density of x, where x is v (vapor) or L (liquid) (kg/m³),
- ρ_A = density of air at STP (kg/m³),
- ρ_W = density of water at STP (kg/m³),
- μ_L = viscosity of liquid (Pa-s),
- μ_W = viscosity of water at STP (Pa-s),
- σ = surface tension (Pa-m), and
- σ_W = surface tension of water at STP (Pa-m).

The factors ψ and λ actually used in MELPROG are calculated from polynomial fits to curves of ψ and λ for steam-water systems versus pressure given in Collier [2.7]. The map that is actually used in MELPROG is a simplified version of the one shown in Fig. 2.2: horizontal stratified flow occurs if $G'_V < 4$ and $G'_L < 2000$; droplet (or annular mist) flow occurs if $G'_V \geq 4$; and horizontal bubbly flow occurs when $G'_L \geq 2000$.

As implemented in MELPROG, both vertical and horizontal flow maps are used in the water-vapor interfacial heat transfer and perimeter calculations. The two maps are interpolated together based on the ratio of the total radial mass flux to the total mass flux. The velocities, differential velocities, and mass flux used in the various Nusselt number correlations are vector magnitudes calculated from the radial and axial components.

An interface sharpener is used which is similar to that in the TRAC code; a horizontal interface is assumed to exist in a cell if the normalized volume fraction of vapor α is < 0.3 in the cell underneath and > 0.7 in the cell above. If these conditions are satisfied, then the horizontal stratified regime is given preference over the vertical regime; otherwise, normal interpolation based on the mass flux ratio, as explained above, is used, which will usually result in a bubbly flow regime for a quiescent pool for cells below the one containing the interface; the cell containing the pool surface will be in horizontal stratified flow. The sharpener is used to select flow regime maps only; the mass fluxes used in the finite difference equations are not altered as is done in Reference [2.1].

2.8.2 Structure Heat Transfer and Drag

Detailed descriptions of the wall shear and wall heat transfer correlations are given in Sections D.1 and D.3,

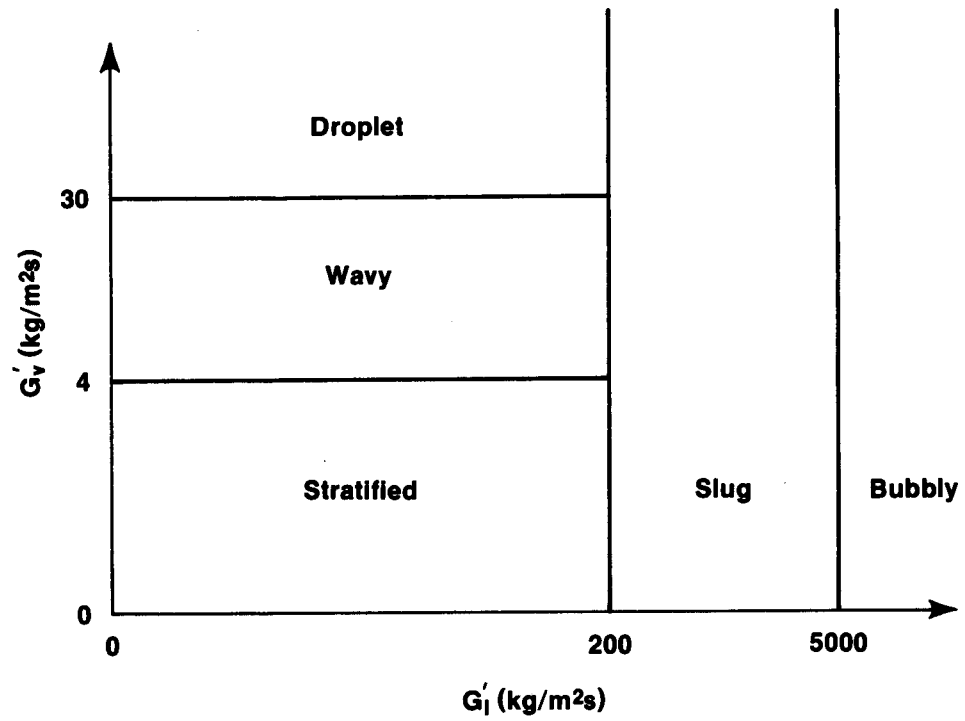


Fig. 2.2 Flow-regime map for two-phase horizontal flow based on work done by Baker [2.6].

respectively, for the vapor and liquid water fields. Heat transfer to the water and vapor fields from vessel structures is done by means of a boiling curve, with additional single-phase flow correlations. There are at present no separate heat transfer coefficients for the individual structures in a cell; an average heat transfer coefficient is used for all structures in a cell. The mass flux and velocities passed to the boiling curve routine are vector magnitudes. The hydraulic diameter is an average of the axial and radial hydraulic diameters. The boiling curve routine, which was based on the one used in TRAC, was reprogrammed, resulting in a much smoother running routine. The major difference between the new heat transfer routine and the original TRAC routine is that additional smoothing between heat transfer regimes is done, so that the heat transfer coefficients will not change abruptly. This new routine was also used as the basis for a new melt heat transfer routine.

2.8.3 Corium Heat Transfer

The corium heat transfer routines incorporate correlations appropriate to an assumed spherical drop (or particle) geometry. These correlations are based on experiments by Dhir and Purohit [2.8]. The heat transfer coefficient (HTC) is plotted as a function of corium temperature in Fig. 2.3 for water at 16 Mpa with $\alpha=0.2$: note that HTC changes by only a small amount except in the film boiling regime (≈ 650 K). The liquid and vapor heat transfer coefficients, HL and HV, respectively, are shown in Fig. 2.4 for water at 16 MPa in the film boiling regime: when α is less than 0.5 or greater than about 0.75, convective heat transfer to only one phase is important; and when α is between 0.5 and 0.75, convection to both phases occurs.

Heat transfer coefficients from the boiling curve and single-phase convective heat transfer coefficients are merged together based on temperature and void fraction to give the final coefficients. Figure 2.5 is a map showing the regions where the different heat transfer coefficients are applicable as a function of temperature and void fraction. Referring to this map, we see that convective heat transfer is used for melt temperatures T_m below the saturation temperature T_g . The convective heat transfer coefficients from the melt to the vapor or liquid (h_v^c and h_l^c , respectively) are formed by first calculating the coefficients as if the surrounding fluid was entirely water or vapor; the coefficients are then adjusted by weighting with the normalized volume fraction,

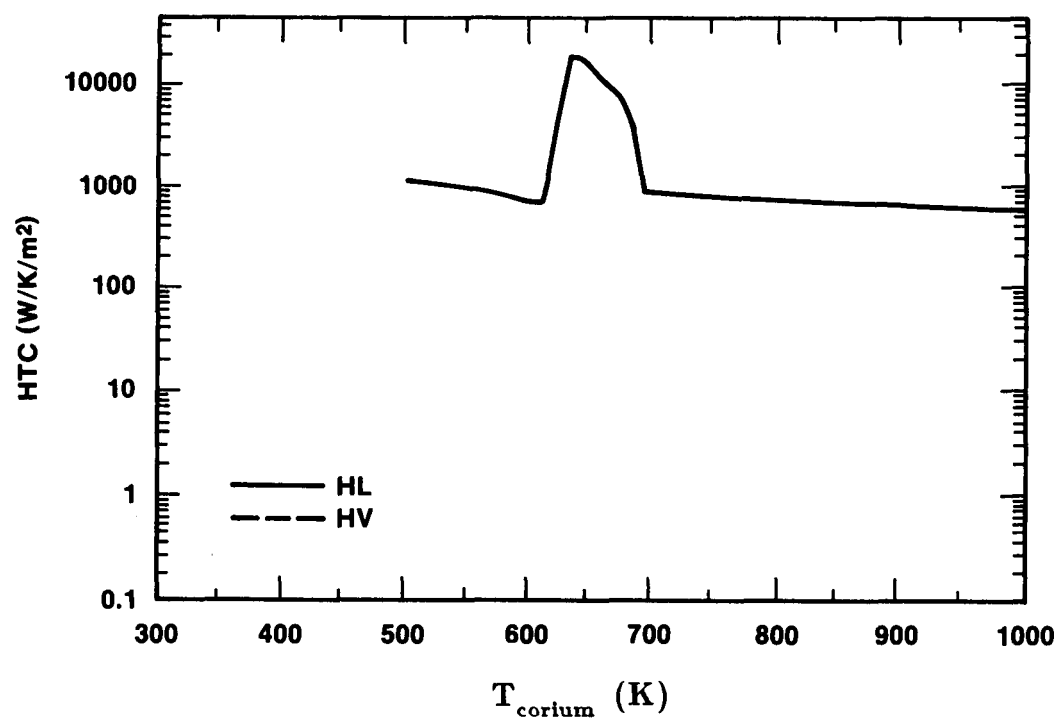


Fig. 2.3 Boiling curve showing the liquid and vapor heat transfer coefficients as functions of temperature.

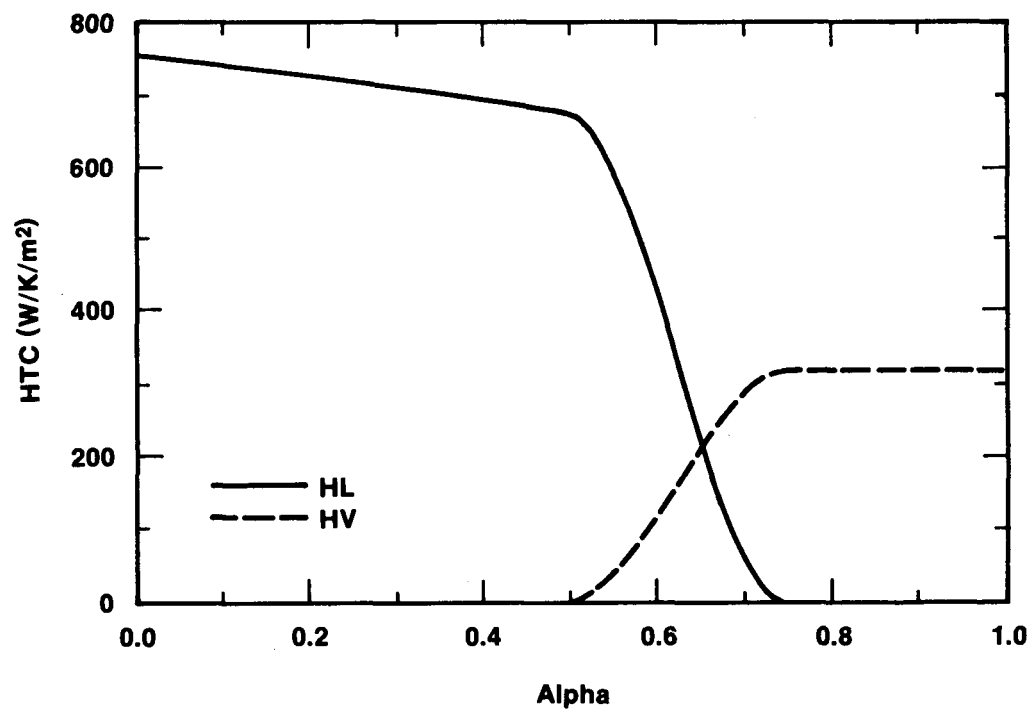


Fig. 2.4 Liquid and vapor heat transfer coefficients as a function of vapor fraction in the film boiling regime.

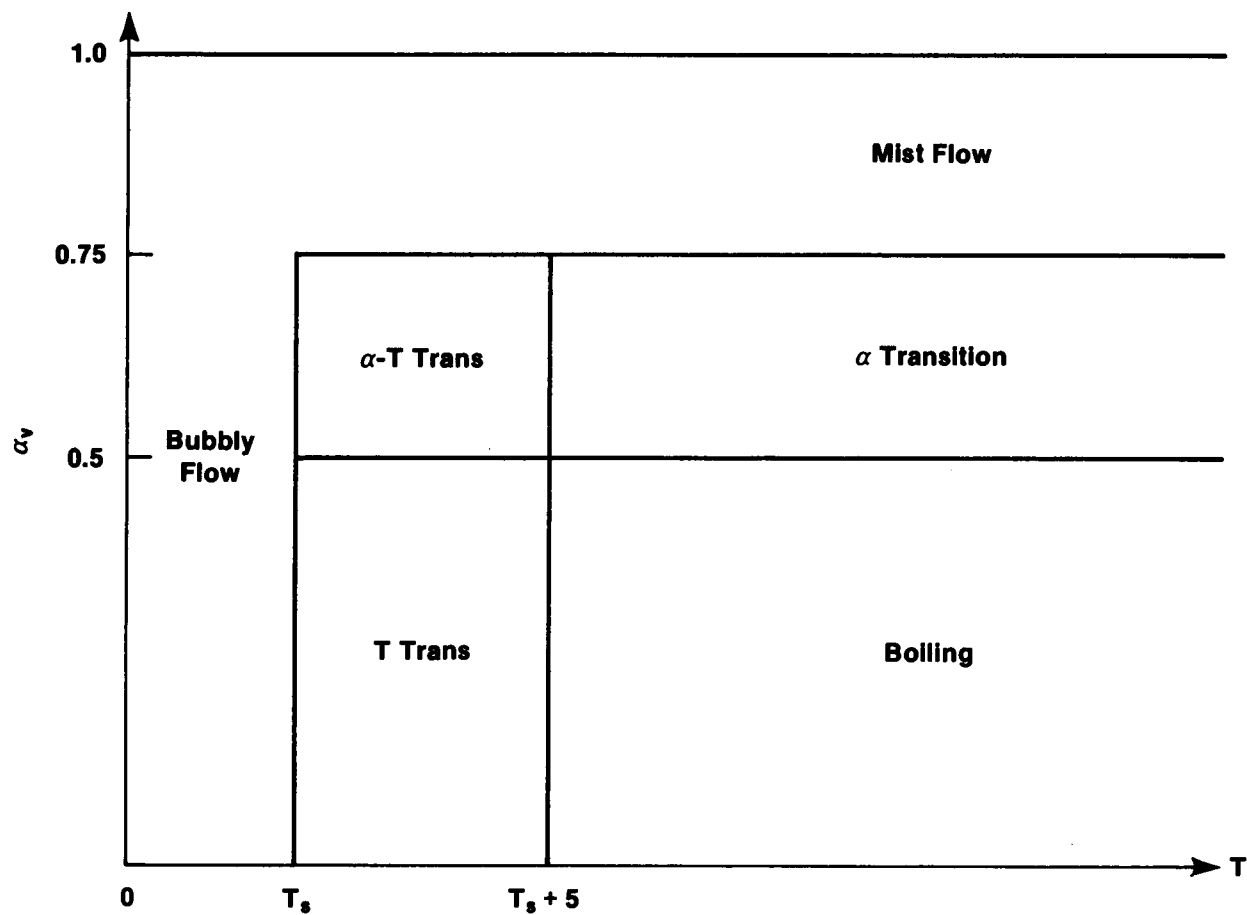


Fig. 2.5 Map showing where the different heat transfer coefficients are applicable as a function of temperature and alpha.

$$h_v^c = \alpha h_v', \quad h_l^c = (1 - \alpha) h_l' . \quad (2.43)$$

As T_m rises above T_s , boiling heat transfer starts to become important. Boiling heat transfer is calculated for $T_m \geq T_s$ and $\alpha \leq 0.75$; above $\alpha = 0.75$, the flow regime is assumed to be pure mist flow, and pure convective heat transfer correlations are used. The pure convective heat transfer regions and the pure boiling heat transfer region are joined by transition regions (see Fig. 2.5); in the transition regions, the convective and boiling coefficients are combined by interpolation in temperature and/or void fraction, so that h_v and h_l are continuous over temperature and void fraction.

The transition region in temperature covers the range $T_s \leq T_m \leq T_s + 5$; the convective and boiling coefficients are joined by linear interpolation in T_m :

$$h_j = c h_j^b + (1 - c) h_j^c ; \quad c = \frac{T_m - T_s}{5} ; \quad (2.44)$$

where the subscript j refers to either vapor or liquid, and the superscript b refers to the boiling curve. The transition in α covers the range $0.5 \leq \alpha \leq 0.75$; over this range, the boiling heat transfer contribution is reduced smoothly to zero as the flow regime changes to pure mist flow at $\alpha = 0.75$. The transition is accomplished by use of cubic interpolating functions for which the derivatives are zero at the endpoints:

$$h_j = f_1 h_j^b + f_2 h_j^c , \quad (2.45)$$

where the interpolating functions f_1 and f_2 are defined as

$$f_1 = (3 - 2x)x^2, \quad f_2 = (3 - 2y)y^2, \quad (2.46)$$

and the terms x and y are given as

$$x = \frac{0.75 - \alpha}{0.25} , \quad y = \frac{\alpha - 0.5}{0.25} . \quad (2.47)$$

The above heat transfer regions cover the domain of interest in MELPROG. A further adjustment is made above the critical point of water, in which case only the convective heat transfer coefficients are calculated.

2.9 Module Interfaces and Boundary Conditions

The FLUIDS module is provided information, and provides information to, the other modules of MELPROG, TRAC, and VICTORIA. This information provides explicit boundary conditions and source terms to the FLUIDS calculation, and also allows implicit linkages with TRAC and VICTORIA. The thermal-hydraulic boundary conditions that are specified on input for stand-alone MELPROG calculations are not actually transferred to FLUIDS from another module, but they are included here for completeness.

2.9.1 Stand-Alone Boundary Conditions

The stand-alone MELPROG vessel requires thermal-hydraulic boundary conditions to simulate interaction with the rest of the RCS. These boundary conditions are of two kinds: velocity (in effect, mass, energy and momentum flow) and pressure. In general, the velocity boundary conditions are used for known (cold leg) flow into the vessel, while pressure boundary conditions are used for unknown hot leg flow. These boundary conditions may be specified at any of the cells on the boundary of the cylindrical segment representing the vessel, with only one kind of boundary condition in a cell. The number of boundary conditions specified is limited only by the number of boundary cells.

The velocity boundary conditions can be specified for liquid water and steam, as functions of problem time. Corium is never allowed to flow into or out of the MELPROG problem space. Also necessary are fluid temperatures and volume fraction, pressure, and hydrogen partial pressure, as functions of problem time, to determine inflow properties.

The pressure boundary conditions are also specified as functions of problem time. In the case of inflow at a pressure boundary condition, the inflow properties are set equal to the properties to the cell adjoining the boundary. For the calculations performed to date, inflow at pressure boundary conditions only occurs during boiling oscillations. The capability for specifying properties for inflow at the pressure boundary condition may be added to the code in the future if warranted.

2.9.2 Interfaces with Other Modules of MELPROG

The FLUIDS module is provided information from the RADIATION, STRUCTURES, CORE, and DEBRIS modules. Most of the information is in the form of volumetric mass and energy source terms ($\text{kg/m}^3/\text{s}$ and $\text{J/m}^3/\text{s}$) that are added to the right hand sides of the appropriate mass and energy conservation equations in both the basic and stabilizer steps. The information that FLUIDS provides the other modules will be discussed in the appropriate sections for those modules.

The RADIATION module provides FLUIDS with volumetric energy sources for each of the fluids on a cellwise basis: these sources account for the absorption of radiation.

The STRUCTURES module provides FLUIDS with volumetric energy sources for each of the fluids on a cellwise basis. It provides: (1) volumetric energy and mass source terms to the corium field when a structural component fails and relocates; (2) volumetric hydrogen mass source terms when a structure is oxidizing, while FLUIDS calculates the related steam sink and energy source term; (3) surface temperatures to allow calculation of heat transfer coefficients on a cellwise basis; (4) the total non-flow volume fraction as the original structural volume fraction in a cell changes (the FLUIDS solution partitions the remaining volume fraction among the fluids); and (5) flow areas and hydraulic diameters on a cell-face basis.

The CORE module provides FLUIDS with volumetric energy sources for each of the fluids on a cellwise basis, from intact and degraded fuel. For degraded fuel, it also provides volumetric mass sources. These sources are further separated into the corium constituents that are tracked in FLUIDS with separate continuity equations. Hydrogen volumetric mass sources are provided during oxidation, while the related steam sink and energy source are calculated in FLUIDS. CORE also provides pin surface temperatures used in the calculation of heat transfer coefficients.

The DEBRIS module provides FLUIDS with volumetric energy sources for each of the fluids, along with volumetric mass sources for corium and hydrogen. Flags are also transmitted (at cell axial faces) that indicate the formation of a stationary bed. At these locations the corium velocities are set to zero.

2.9.3 Link to TRAC

MELPROG can be run as a "MELVSL" component in the TRAC component network of the RCS of a PWR. This is possible because both the FLUIDS module and TRAC utilize the SETS method to solve the two-fluid problem. In this configuration MELPROG replaces the normal TRAC vessel component (the TRAC model considers the motion of only two fields, vapor and coolant). The link is only possible with TRAC/PF1-MOD2, which already has in place the logic necessary for coupling to a SETS-solved multidimensional vessel.

The linked code has a structural hierarchy in which MELPROG is controlled by TRAC, with MELPROG treated as a component in the TRAC system network.

2.9.3.1 Model Limitations

The modeling limitations are:

- a. Only one MELVSL component is allowed in a system.
- b. The common block MELHOOK, which contains most of the coupling information between the two codes, has hardwired dimensions. These dimensions reflect the current maximum values of
 - the number of MELPROG-TRAC junctions (25).
 - the number of MELPROG-TRAC junctions per TRAC loop (3).
 - the number of MELPROG-TRAC junctions per MELPROG cell (4).

These dimensions can be changed easily if necessary.

- c. Momentum coupling from the loop components is only to the radial momentum equations in MELPROG. This is true for typical PWR configuration, but not for boiling-water reactors (BWRs). This limitation will be removed in conjunction with the development of a BWR jet pump model.

The SETS numerics require that a number of implicit systems of equations be solved in MELPROG/TRAC. That is, new time variables in one component are expressed as linear functions of new time variables in other components. The variables solved for are the two fluid stabilizer velocities, pressure corrections, the two fluid mass densities, the two fluid energy densities, and hydrogen mass densities. Only the radial velocities in MELPROG are coupled to TRAC, so that the number of implicit systems of equations linking TRAC and MELPROG is eight. The solution of these systems provides a tight coupling between the two codes and is the key to understanding the link.

These systems of equations are partitioned as in TRAC. For clarity, the case of two MELPROG/TRAC junctions will be discussed here. Each system is of the form:

$$\underline{A}\vec{x} = \vec{b}$$

where

$$\underline{A} = \begin{bmatrix} \left[\begin{array}{c} \underline{V} \end{array} \right] & a_{1j} \quad a_{2k} \\ a_{j1} \quad a_{k2} & \left[\begin{array}{c} \underline{J} \end{array} \right] \end{bmatrix} ; \vec{x} = \begin{bmatrix} [x_1] \\ [x_2] \end{bmatrix} ; \vec{b} = \begin{bmatrix} [b_1] \\ [b_2] \end{bmatrix} . \quad (2.48)$$

The matrix $\underline{A}$ in Eq.(2.48) is composed of diagonal blocks $\underline{V}$ and $\underline{J}$, plus a maximum of four off-diagonal non-zero elements. $\underline{V}$ represents the coupling of cells within the vessel and is, in general, a banded matrix with five stripes with a limited number of outliers representing coupling by the embedded flow paths in MELPROG. The block $\underline{J}$ represents coupling between the TRAC network junctions that do not connect to MELVSL. This block is formed after the solution by cell within each component in terms of the junction variables. The nonzero off-block-diagonal elements represent the coupling between MELVSL and the junction variables. The coefficient a_{1j} represents the coupling coefficient between MELPROG junction cell 1 and network junction j . The coefficient a_{2k} represents the coupling between MELPROG junction cell 2 and network junction k . The coefficients a_{j1} and a_{k2} represent similar couplings in the opposite directions.

The partitioned system is solved as follows: first, the following system is solved:

$$\underline{J}\vec{x}_2 = \vec{b}_2 + XV_1 \cdot \overline{CV}_1 + XV_2 \cdot \overline{CV}_2 . \quad (2.49)$$

The block $\underline{J}$ is the coefficient matrix, while the coupling to MELVSL is treated as unknown vectors on the right hand side (RHS) of the equation. Vectors CV_1 and CV_2 have only one nonzero component, $-a_{j1}$ and $-a_{k2}$ respectively. XV_1 and XV_2 are the unknowns at MELPROG junction cells 1 and 2, respectively (elements of $\vec{x}_1$). The solution to this system is given by:

$$\vec{x}_2 = \vec{b}_2' + XV_1 \cdot \overline{CV}_1' + XV_2 \cdot \overline{CV}_2' , \quad (2.50)$$

and

$$\underline{V}'\vec{x}_1 = \vec{b}_1' . \quad (2.51)$$

Included in Eq.(2.50) are linear relationships for X_j and X_k in terms of XV_1 and XV_2 . These relationships are substituted into the solution vector $\vec{x}_2$ in Eq.(2.48), effectively eliminating a_{1j} and a_{2k} and modifying $\vec{V}$ to $\vec{V}'$ and $\vec{b}_1$ to $\vec{b}'_1$. The MELVSL component is thus numerically decoupled from the junction variables, and Eq.(2.51) may be solved for $\vec{x}_1$. After this system is solved, XV_1 and XV_2 can be taken from the solution vector $\vec{x}_1$ and back substituted using Eq.(2.50) to give $\vec{x}_2$. $\vec{x}_1$ and $\vec{x}_2$ represent the solutions within the vessel and at the network junctions, respectively. The solutions within each component are found by further back substitution (the component variables are expressed as linear functions of the junction solution vector $\vec{x}_2$). Equation (2.49) is solved in TRAC, and the information represented by Eq.(2.50) is transported to MELPROG, where Eq.(2.51) is solved. The solution vector of Eq.(2.51) is then transferred back to TRAC, where the back substitution of Eq.(2.50) is accomplished. This procedure has been described for the case of two MELPROG-TRAC junctions. More than two junctions generate more off-block-diagonal elements in Eq.(2.48), and more RHS vectors in Eqs.(2.49) and (2.50).

This procedure is used for each of the eight systems of equations listed above. For the pressure correction equation, additional work is necessary to calculate the coefficients a_{1j} and a_{2k} . For each mesh cell in MELPROG, the following equation must be solved for each iteration during the basic step:

$$\underline{B}\vec{y} = -\vec{S} - \sum_{i=1}^5 \vec{c}_i \delta p_i \quad . \quad (2.52)$$

Equation (2.52) represents the basic Newton iteration on changes in pressure, volume fraction and temperature necessary to balance the mass and energy conservation equations. In Eq.(2.52), $\underline{B}$ is the Jacobian matrix of first derivatives, $\vec{y}$ is the solution vector (composed of changes in pressures, temperature, and volume fractions), $\vec{S}$ is the source vector, and the summation is over the given cell and four neighboring cells (the pressure change in $\vec{y}$ is for compressibility only). The lengths of the vectors and size of the matrix are determined by the number of fluids present in the cell. In MELPROG/TRAC another term must be added to the right hand side, giving:

$$\underline{B}\vec{y} = -\vec{S} - \sum_{i=1}^5 \vec{c}_i \delta p_i - \sum_{ii=1}^{n_{jun}} \vec{cs}_{ii} \nabla(\delta p_{jun})_{ii} \quad . \quad (2.53)$$

Here $V\delta p_{jun}$ is the change in pressure drop across the junction that connects to the given MELPROG cell. The solution to Eq.(2.53) is given by:

$$\vec{y} = -\vec{S}' - \sum_{i=1}^5 \vec{c}'_i \delta p_i - \sum_{ii=1}^{n_{jun}} \vec{cs}'_{ii} V(\delta p_{jun})_{ii} \quad (2.54)$$

The first element in $\vec{y}$ is δp , so the first row of Eq.(2.53) is a linear relationship between pressure changes, and the coefficient of $V\delta p_{jun}$ is either a_{1j} or a_{2k} . Eq.(2.53) is solved in subroutine JAC41H.

2.9.3.2 Calculation Control

TRAC has overall control of the calculation, using information from MELPROG. Both TRAC and MELPROG input decks are read, and both sets of output files are written. The MELPROG input deck is identical to the stand-alone deck, with the exception that no boundary conditions are read in. The TRAC input deck has a few lines to identify the MELVSL component.

TRAC controls the time-step size, with a maximum time-step provided by MELPROG. The MELPROG maximum time step is based on a maximum Courant number or accuracy conditions on rate of change of temperature or volume fraction. TRAC also controls the edit and dump intervals, along with the calculation end time. MELPROG will give edits and dumps at the times determined by TRAC.

TRAC calls MELPROG from five subroutines. Subroutine RMPROG is called from INPUT to read the MELPROG input files and perform some initialization. Subroutine CIVSLM is called from ICOMP to perform more initialization. This call replaces the call to CIVSSL, the standard TRAC vessel initialization routine. During the calculation, PREP3DM, OUT3DM and POST3DM are called from PREP, OUTER, and POST, respectively, replacing calls to PREP3D, OUT3D, and POST3D (the last letter M in general denotes a MELPROG subroutine or variable).

The subroutines CIVSLM, PREP3DM, OUT3DM, and POST3DM are coupling routines, that is, they both have TRAC and MELPROG COMMON BLOCKS, and can set the variables in COMMON MELHOOK. All communication between the two codes (except for the BD array described below) is by variables in COMMON MELHOOK, which is the

only COMMON BLOCK that occurs in both codes. Although this strategy entails more information transfer, it is necessary to keep the bookkeeping manageable between the two dynamic codes.

The coupling routines CIVSLM, PREP3DM, OUT3DM and POST3DM call subroutine FLDYN in MELPROG. The variable IENTRY determines the function to be performed in MELPROG. The possible values of IENTRY are given in Table 2.3. Subroutine J3DM is another coupling subroutine that performs a function similar to J3D in TRAC. Using information from MELPROG COMMON BLOCKS, it sets the boundary condition array BD that is used by the other components in TRAC. The difference in the definition of volume fraction between the two codes is accounted for J3DM.

The TRAC logic for backing up to the beginning of a time step will not be described in detail here, except to note the following coupling with MELPROG. The TRAC variable IIFAIL, which was once used to denote inner iteration failure, is now used to denote that MELPROG has calculated a negative volume fraction, and that backup should occur. MELPROG resets variables on backup in subroutine TSTEP.

2.9.4 Link to VICTORIA

A link between VICTORIA, MELPROG and TRAC will allow simulation of fission product release from fuel, transport and deposition in the RCS, and release into containment. The link consists of three main parts: first, the transfer of information from MELPROG and TRAC to VICTORIA to allow the solution of local equilibrium chemistry, fuel rod diffusion, and aerosol behavior; second, the feedback of decay power source terms to MELPROG and TRAC; and third, the solution of the continuity equations for the gaseous fission products and aerosols in the vessel and the balance of the RCS.

Currently, steps one and two have been completed for the link to MELPROG, but not to TRAC. Step three has been completed. The solution for step three utilizes the linked-continuity equation solution of the mass stabilizer. The gas transport matrix is used, with each fission product species requiring a different right hand source vector. Thus only a single matrix decomposition step is required, and the multiple back substitutions can be efficiently vectorized on the number of right hand sides.

Table 2.3		
Definition IENTRY Flag		
VALUE OF IENTRY	CALLED FROM (TRAC)	FUNCTION IN MELPROG
1	ICOMP	Initialization
2	Not used	
3	PREP3DM	Solve momentum equations Evaluate explicit coefficients
4	OUT3DM	Evaluate implicit coefficients Newton iteration solve Eq.(2.52) or (2.53) within each cell, evaluate coefficients for pressure-correction equation (Eq.(2.51)).
5	OUT3DM	Back substitute for changes in volume fraction and temperature, within each cell (Eq.(2.54))
6	POST3DM	Solve mass and energy stabilizer equations
7	POST3DM	Call other modules in MELPROG

The link between TRAC and MELPROG is both simple and modular, because MELPROG simply replaces a component in the TRAC network. Unfortunately, it was not possible to extend this simplicity to the link between TRAC and VICTORIA; it was necessary to make additions to many lower level TRAC subroutines. While these additions are clear in purpose (they simply extend an already present networking solution of the "air" mass stabilizer), their sheer number will require upkeep in the future.

2.10 Fluids Nomenclature

a, b coefficients in linearized momentum equations
C, D inter-fluid and wall friction factors
F flow area
g gravitational acceleration

h	enthalpy
P	pressure
Q	volumetric heat exchange rate
R	radial direction
t	time
T	temperature
u	internal energy
U	radial velocity
$\bar{U}$	radial predictor velocity
$\bar{\bar{U}}$	radial stabilizer velocity
$\vec{V}$	two-component velocity vector
W	axial velocity
$\bar{W}$	axial predictor velocity
$\bar{\bar{W}}$	axial stabilizer velocity
VOL	cell geometric volume
Z	axial direction

Greek

α	volume fraction
Γ	volumetric mass exchange rate
ρ	microscopic density

Subscripts

f	fluid index
h	hydrogen
i	radial node index
j	axial node index

Superscripts

n	time level

2.11 References

- 2.1 Safety Code Development Group, TRAC-PF1, An Advanced Best-Estimate Computer Program for Pressurized Water Reactor Analysis, NUREG/CR-3567, LA-9944-MS, Los Alamos National Laboratory, Los Alamos, NM (1984).
- 2.2 J.H. Mahaffy, "A Stability-Enhancing Two-Step Method for Fluid Flow Calculations," Journal of Computational Physics **46**, No. 3 (June 1982).
- 2.3 P.J. Maudlin, E.C. Schwegler, T.L. Wilson, J.L. Tomkins, A.W. Cronenberg and J.A. Moore, A Damage Assessment of TMI-2, LA-UR-83-737, Los Alamos National Laboratory, Los Alamos, NM (1983).

- 2.4 S.V. Patankar, Numerical Heat Transfer and Fluid Flow, McGraw-Hill (1980).
- 2.5 R.J. Lipinski, "A Coolability Model for Postaccident Nuclear Reactor Debris," Nuclear Technology 65, pp. 53-66 (1984).
- 2.6 O. Baker, "Design of Pipe Lines for Simultaneous Flow of Oil and Gas," Oil and Gas Journal (1954).
- 2.7 J.G. Collier, Convective Boiling and Condensation, McGraw-Hill, pp. 18-20 (1972).
- 2.8 V.K. Dhir and G.P. Purohit, "Subcooled Film-Boiling Heat Transfer from Spheres," Nucl. Eng. and Design 47, pp. 49-66 (1978).

3.0 CORE MODULE

The goal of the MELPROG project is the development of a mechanistic computer model for analysis of in-vessel phenomena that can occur during severe accidents in light water reactors (LWRs). Achievement of this goal implies the ability to treat intact geometry and the path from intact geometry through failure of the reactor vessel and movement of core material out of the vessel. It also implies an ability to model and analyze experiments designed to examine relevant physics. To meet these goals a detailed core degradation model, both in terms of geometry and in terms of phenomena treated, is necessary. These goals can only be met when the level of modeling is sufficiently detailed that the calculated results do not change significantly as a function of modeling uncertainties. Determination of the modeling uncertainties is of necessity an iterative process for a large integrated systems code like MELPROG. The procedure is to apply the code to a variety of accident scenarios and to severe accident experiments while evaluating individual models and the integration of models to determine deficiencies. Then, models and the coupling of models are modified where indicated by the calculational results and where the available data base exists to improve the models and the calculations are repeated. This procedure is also useful for defining the need for additional experimental data and for phenomenological models.

The PINS module of MELPROG-PWR/MODO explicitly treated fuel rods only. It treated only one representative fuel rod per radial ring. Control rods and poison rods were modeled as mass sources for debris formation only. PINS modeled intact fuel rods, fuel rods undergoing oxidation, and fuel rods ballooning and bursting. The initial fuel rod geometry could only be treated as uniform axially. Grid spacers, fission gas plenums, end plugs, etc. could not be modeled. (Grid spacers are of particular importance because of their potential interaction with fuel rod cladding and control rods and because of the role grid pacers may play in relocation of core material and the formation of in-core debris beds as indicated by the DF experiments.) As a result it was judged that modifying PINS to remove the many deficiencies would be difficult at best and more than likely lead to an unsatisfactory result. Therefore, it was decided to construct a new module, CORE, that would be designed to treat in a consistent and flexible framework all PWR and BWR core structures.

3.1 General Description

CORE is designed to treat two classes of structures, those that can be modeled as a cylinder or group of concentric cylinders and those that can be modeled as a one or more layered slab. Fuel rods, PWR control rods, poison rods, and dummy rods are examples of core structures that can be treated with cylindrical geometry while BWR can walls and control blades are examples of core structures that can be modeled in slab geometry. In general, any structure whose primary extent is axial and whose major heat transfer direction is radial can be treated. The actual level of geometric detail, both axially and radially, is determined by the user. He is limited only by his available computer resources. The geometry of all core structures can change as a result of melting, crusting, oxidation, etc.

Conduction heat transfer is one-dimensional in the radial direction. (Two-dimensional conduction heat transfer may be included in future versions of CORE.) Axial coupling is through convective and radiative heat transfer. The solution method used is a modified Crank-Nicolson [3.1] formulation with variably weighted forward-backward differencing. Forward weighting is always less than 0.5. Phase changes, melting and freezing, are treated implicitly and the heat of fusion energy is rigorously conserved. A conduction heat transfer limited freezing model is used to calculate crust formation. Steady state calculations are performed by using a time step of 10^6 s.

Chemical changes that are treated by core include oxidation of zircaloy cladding on inner and outer surfaces, oxidation of liquid zirconium in contact with a ZrO_2 layer through which oxygen is diffusing, and oxidation of U-Zr-O mixtures flowing on the exterior of core structures. Also, oxidation of stainless steel and B_4C are treated. Additional chemical changes that are treated include formation of U-Zr-O mixtures, dissolution of ZrO_2 by liquid Zr, attack on zircaloy by inconel and subsequent attack on the stainless steel cladding of control rods, Ag-In-Cd control material attack on zircaloy, and foaming of stainless steel during oxidation.

Energy deposition in core structures is considered from chemical and nuclear sources. Oxidation is the major source of chemical energy, however, heats of formation for other chemical reactions are considered. Fission energy is considered in the form of power as a function of time while decay heat can be

treated as power as a function of time or determined from the fission product inventory as a function of time. (Fission product inventory is determined by VICTORIA - see Section 7.) Each core structure has its own power factor for decay and fission heating.

Mechanical changes that are treated include thermal expansion, fuel swelling, ballooning/collapse, bursting, and crack formation. Cladding failure can occur by bursting, by accumulated damage, by dissolution, and by melting. Accumulated damage failure is based on a linear life fraction rule. Crack formation is used to determine flow paths for liquid material and fission products out of fuel rods.

When core structure materials liquefy they are allowed to flow axially on solid core structure exterior surfaces and in gaps that can exist between two solid regions in a core structure. This flow, candling, is driven by gravity, friction and capillary forces. Axial flow paths are coupled when cracks or holes exist in a solid region separating two flowing regions. The flow solution is incompressible and conserves mass, momentum, and energy. All core structures can have candling in gaps and on the innermost and outermost solid surfaces.

CORE uses a dynamic mesh to treat changes in geometry that occur because of chemical reactions, phase changes, and convection of material. Initially each structure consists of one or more material regions in the radial direction. During remeshing, an additional material region can be treated on each surface of each region. Regions can also lose all of their mass. After all mass changes have been completed for each region the individual regions are examined to determine if they should be combined or deleted. Regions that have lost all mass are deleted while regions that are in contact with regions of the same material are combined into a single region. This general procedure allows for tremendous flexibility in the ways that a core structure's geometry can change during an accident calculation.

CORE interacts with the FLUIDS module through energy and mass transfer terms and through changes in flow areas and hydraulic diameters. CORE provides fuel rod thermal and geometric data to VICTORIA for fission product release calculations. VICTORIA provides CORE with decay power in fuel rods and plated out on surfaces. CORE provides geometry and

thermal data to RADIATION and receives surface heat fluxes in return. CORE provides geometry information to STRUCTURES for use in the calculation of structure volume fractions, flow areas, and hydraulic diameters. CORE also interacts indirectly and directly with DEBRIS. Its indirect interaction is through the FLUIDS module: CORE can pass materials to FLUIDS which may then pass that material to DEBRIS to form beds. If a debris bed forms in the core region, CORE and DEBRIS interact directly via mass and energy sources and sinks.

3.2 Detailed Description of CORE

3.2.1 Geometry

CORE is designed to treat structures whose primary extent is axial and whose major heat transfer direction is radial. Each structure's location is determined by the radial ring it is in and by its lowest point relative to the problem bottom. These conventions allow for structures to be partially in the core region, for example, control rods that are not fully inserted. Also, the axial dimension of each structure is independent of the other structures. Thus, partial length rods can be treated.

All core structures are modeled as consisting of one or more axial regions. An axial region is defined as a section of a core structure that has a uniform radial geometry, initially. Each region can have one or more axial meshes. To illustrate the concept of an axial region, consider a PWR fuel rod with a bottom end plug, a bottom insulator region, a fissile region, a top insulator region, a fission gas plenum, and a top end plug. Each of these axial sections could be modeled as an axial region resulting in a total of six axial regions in one fuel rod model. Figure 3.1 shows a typical fuel rod model.

Each axial region is composed of one or more radial regions. A radial region consists of a single material that initially extends axially for the length of the axial region. There are six types of radial regions: type 1 is a manufactured solid region, type 2 is a manufactured solid region with holes in the radial direction, type 3 is a created solid region that can have holes that extend radially, type 4 is a manufactured powder region, type 5 is a region in which material can flow between two solid regions, and type 6 is a region in which material can flow on one solid surface. Region types 1, 2, and 4 can be input by the user while region types 3, 5, and 6 are created by CORE in

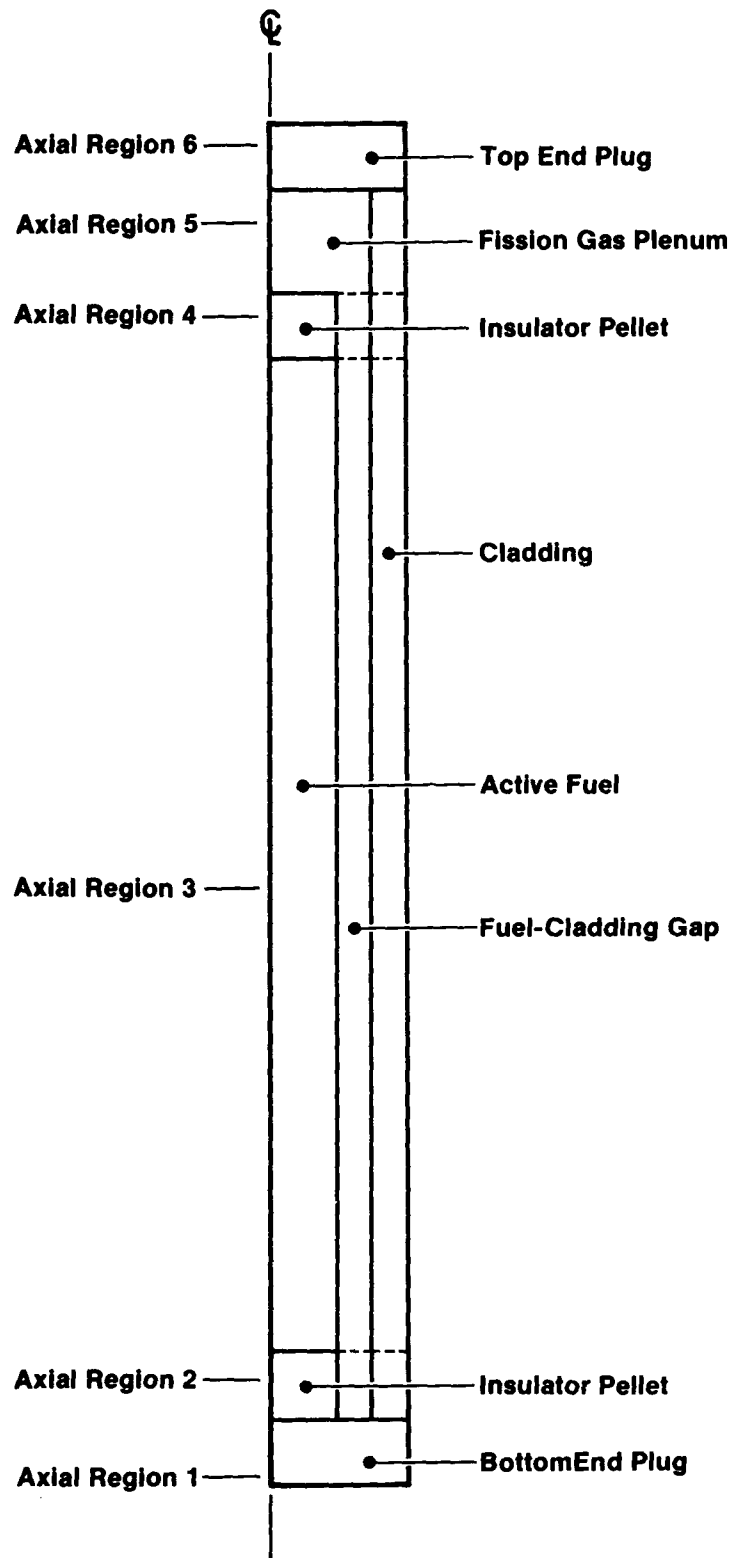


Fig. 3.1 Typical fuel rod model. (no scale)

response to changes in the geometry of a structure. (Details of the formation and loss of regions are given in Section 3.2.7.) In Figure 3.1 the radial regions of a typical fuel rod model are shown. As indicated, a gap exists between the fuel and the cladding. CORE allows gaps to be input between any two input regions.

Within an axial region the mesh size is limited to a user specified maximum. The actual mesh is determined by the relationship of the core structure to the fluids axial mesh. Core structure meshes are always contained within a fluids mesh. For example, consider a 4.0 m long active fuel section of a fuel rod. Also, consider that the FLUIDS axial mesh is set at 1.0 m and the maximum desired fuel rod mesh is 0.3 m. For simplicity, assume that the bottom of the active fuel coincides with the bottom of a FLUIDS axial mesh. Then the fuel rod axial mesh is 0.25 m and there are 16 axial meshes in the fuel rod over a distance of 4.0 m.

CORE's flexibility to treat a wide variety of LWR core structure geometries is best explained by considering examples of how the major PWR and BWR core structures might be modeled. First, consider the fuel rod shown in Fig. 3.1. As shown, the fuel rod appears to consist of 6 axial regions. These are, starting from the bottom, end plug, insulator pellet, active fuel, insulator pellet, fission gas plenum and end plug. However, if the grid spacers are treated as part of the fuel rod as shown in Figure 3.2, there are then 18 axial regions. These are, starting from the bottom, end plug, insulator pellet, active fuel, active fuel with grid spacer, active fuel, active fuel with grid spacer, active fuel, active fuel with grid spacer, active fuel, active fuel with grid spacer, active fuel, insulator pellet, fission gas plenum and end plug. When grid spacers are modeled in this way their heat capacitance and mass are accounted for. In addition, flow blockage effects and chemistry between incaloy and zircaloy are treated by CORE.

As a second example of the geometric flexibility of CORE, consider a simulated fuel rod for use in an out-of-pile experiment. This rod is shown in Figure 3.3. It has 3 axial regions, insulator, heated section and insulator. In the radial direction there are 3 material regions and 2 gaps. The innermost region is a resistance heating element, then there is a gap, a fuel region, and another gap, and the final region is the

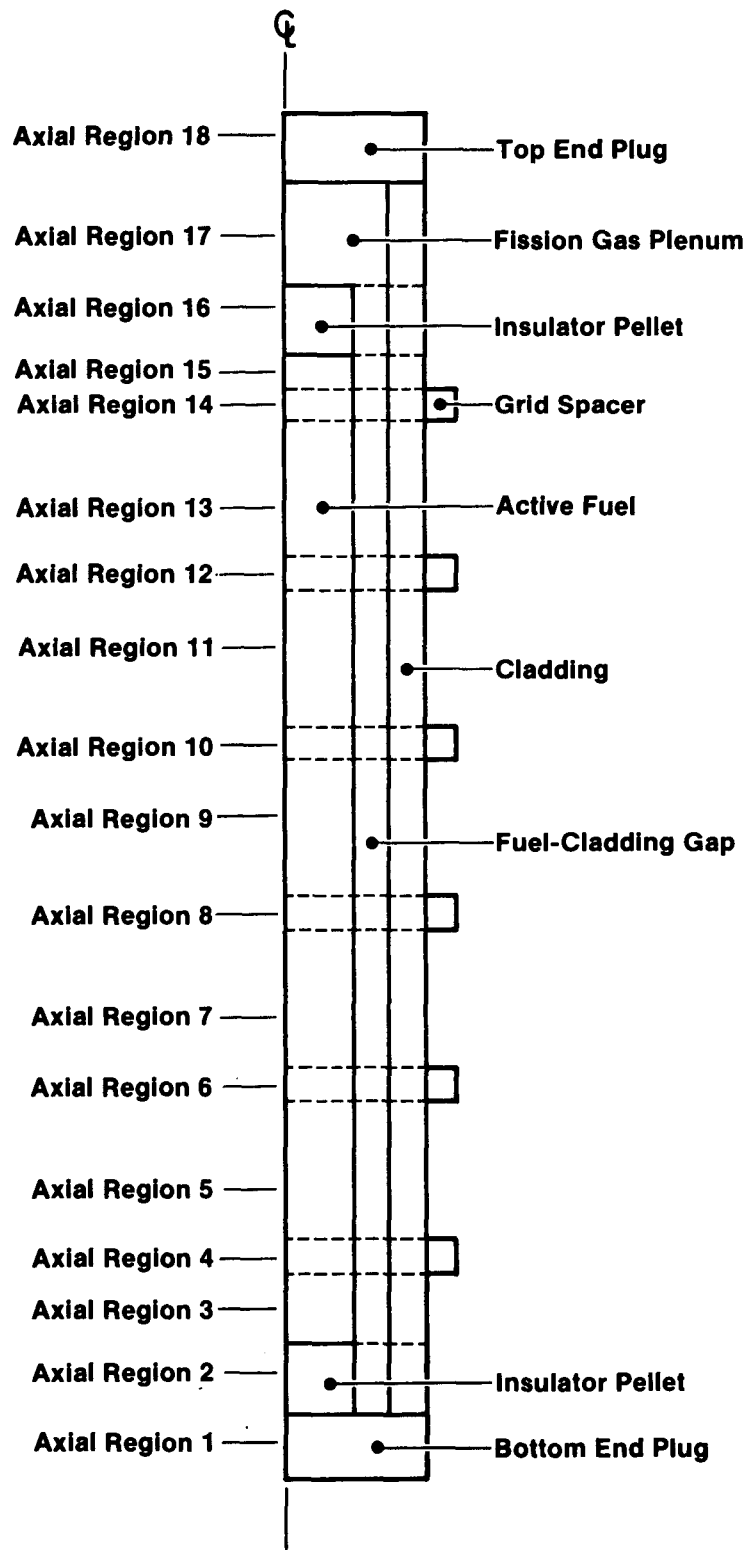


Fig. 3.2 Typical fuel rod model with grid spacers.
(no scale)

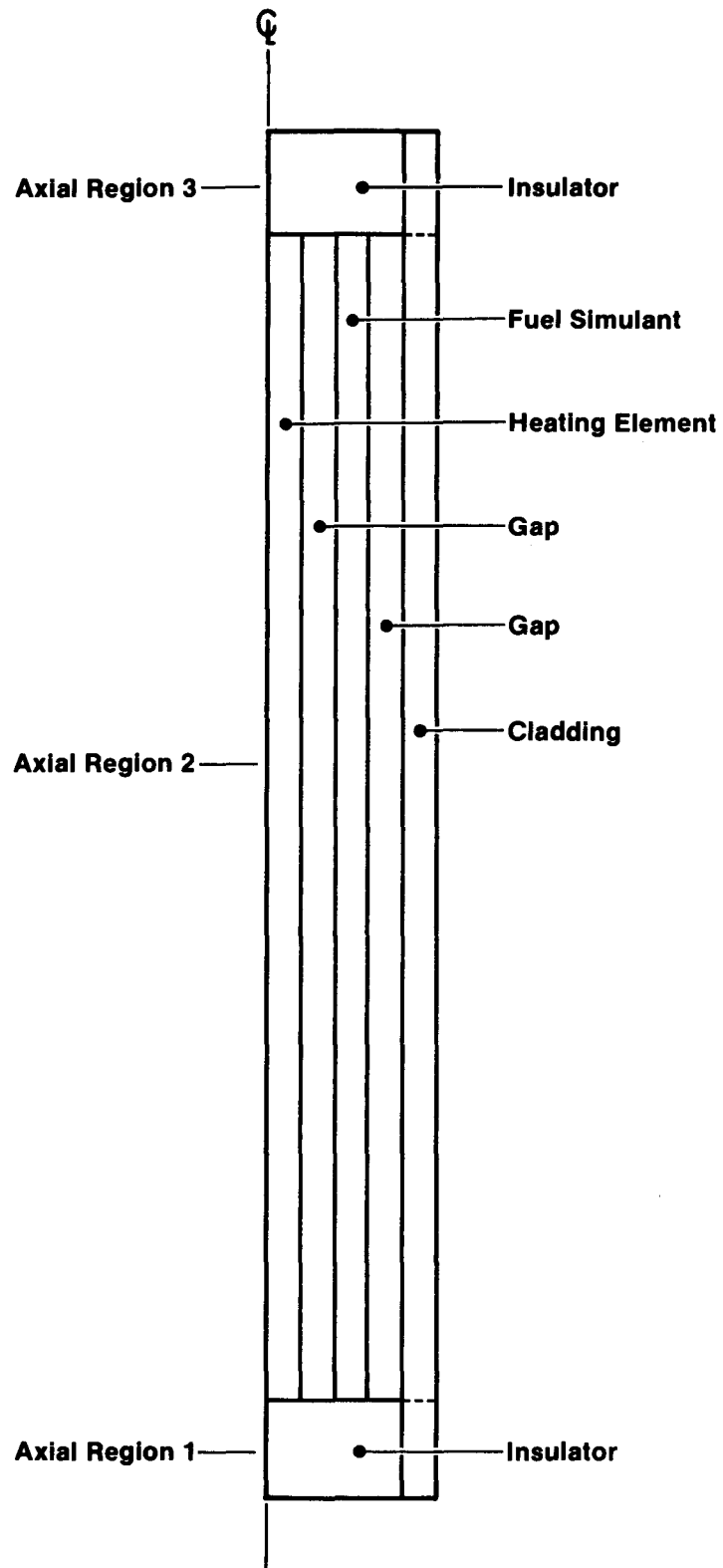


Fig. 3.3 Fuel rod used in an out-of-pile experiment.
(no scale)

cladding. As in the first example, grid spacers could be modeled by adding a fourth material region at appropriate axial locations.

A third example, that illustrates the flexibility of CORE in modeling various geometries, is a model of a BWR control blade. As shown in Figure 3.4 a BWR control blade is shaped like a cruciform. It has an outer sheath of stainless steel. The sheath has holes that allow cooling water pumped into the control blade at its bottom to flow out into the channel between can walls. The neutron absorber material (B_4C) is a powder contained in stainless steel tubes. To reduce the impact of compacting of the B_4C powder stainless steel blockages are located in the tubes at approximately 0.25 m apart over the length of the control rod. Because CORE models only cylinders and slabs it is necessary to treat a BWR control blade as either a slab or a cylinder but not both. The overall geometry of a BWR control blade more closely resembles a slab than a cylinder, therefore, a slab treatment is chosen for this example. A cross section of a possible model for a BWR control blade is depicted in Figure 3.5. As shown, the model consists of 5 "radial" material regions. Region 1 is the stainless steel sheath. Next, there is a gap. Region 2 represents the stainless steel cladding of the absorber rods. Region 3 represents the absorber material. Region 4 represents the stainless steel cladding of the absorber rods. There is then a second gap and, finally, region 5 which is the stainless steel sheath. Treating a BWR control blade in this manner provides for heat transfer through the blade (radial direction) and is consistent with the two-dimensional (R-Z) geometry of MELPROG. Also, this model allows the mass and thickness of each region to be conserved. As in the fuel rod example, the first example above, the stainless steel blockages in the B_4C rods can be treated by using multiple axial regions.

3.2.2 Core Heat Transfer

The technique used to solve the heat equation is described in the following section. Bulk melting and freezing is discussed in Section 3.2.2.2 and conduction limited freezing is considered in Section 3.2.2.3.

3.2.2.1 Conduction Heat Transfer

Conduction heat transfer is calculated using a modified Crank-Nicolson finite difference technique [3.1] for material nodes and a fully implicit technique for surface nodes. Material

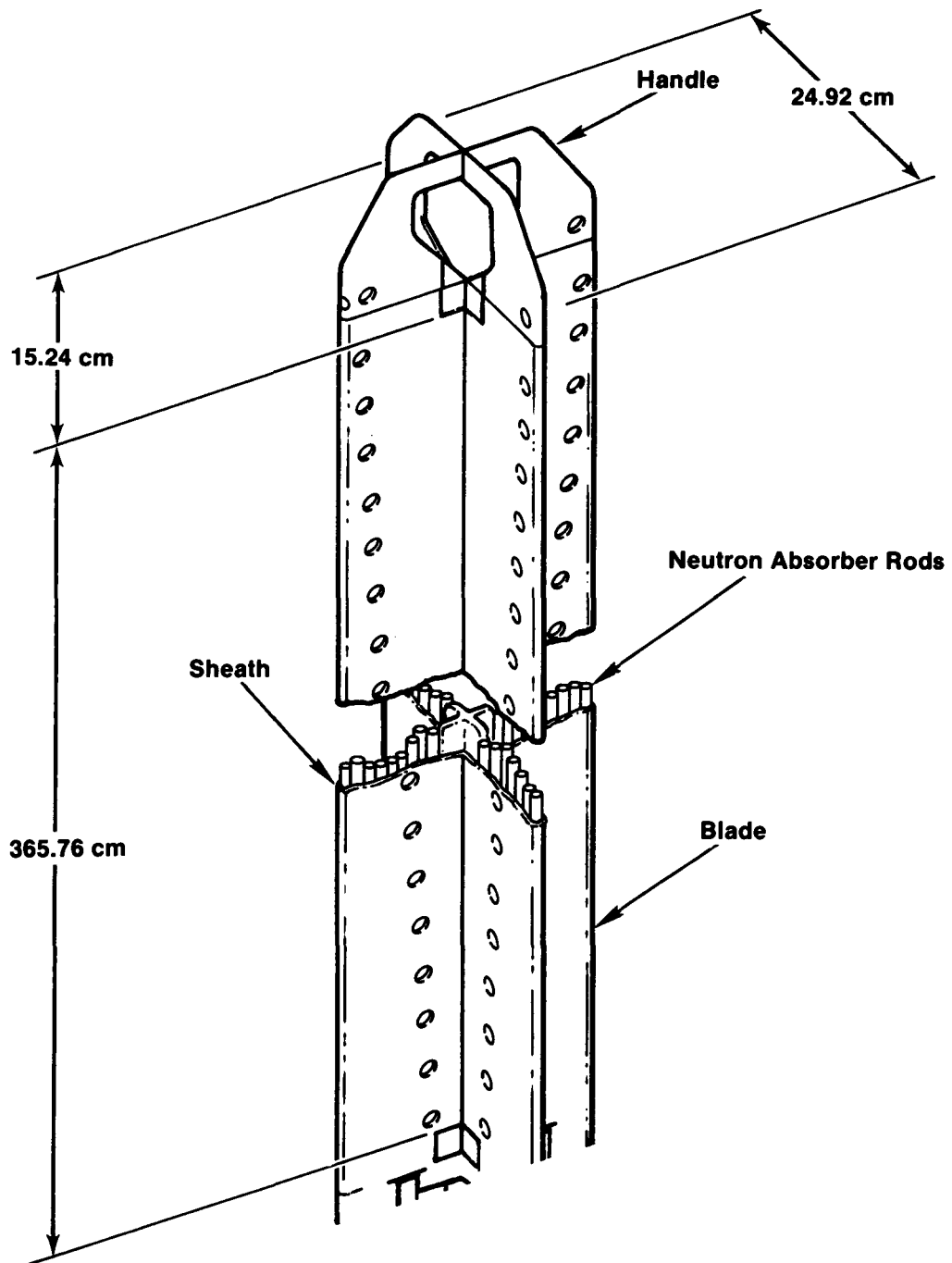


Fig. 3.4 BWR control rod assembly.

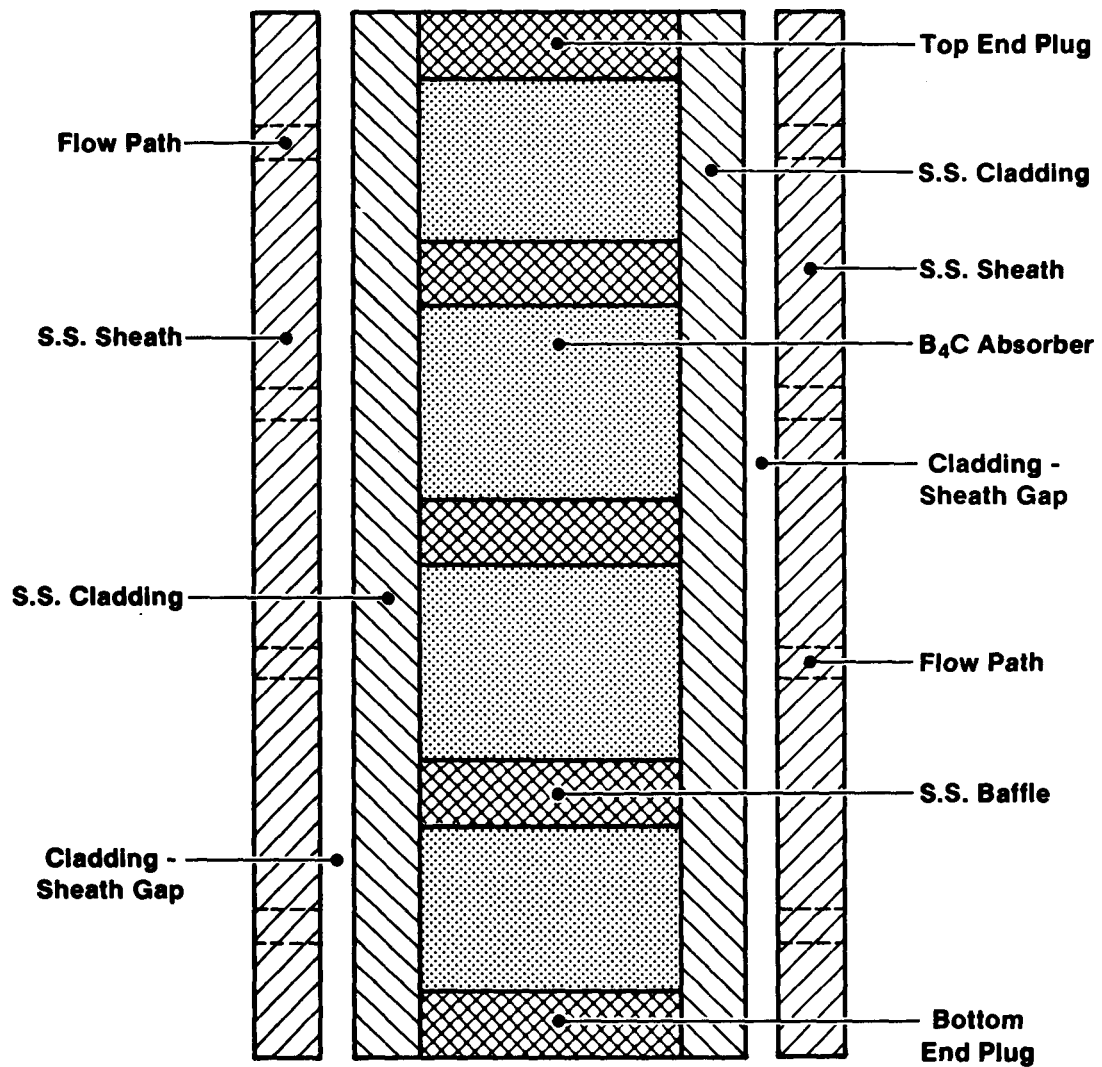


Fig. 3.5 BWR control blade model. (no scale)

nodes have mass associated with them while surface nodes do not. The characteristics of each type of node are given in Table 3.1. An example of the arrangement of material and surface nodes in a radial cross section of a typical fuel rod model is shown in Figure 3.6. As shown, there are surface nodes at the fuel rod center, on the active fuel outer surface, on the inner surface of the cladding, at the interface between the cladding and grid spacer, and on the outer surface of the grid spacer. (See Figure 3.2 and Section 3.1 for an example of how grid spacers might be modeled with CORE.) Five material nodes are shown in the fuel, three are shown in the cladding, and one is shown the grid spacer. Initially, within each region (fuel, cladding, and grid spacer), the material nodes represent equal masses. That is, the fuel mass is divided into five equal mass nodes, the cladding mass is divided into three equal mass nodes and the grid spacer mass is one node. During melting, oxidation, etc., the mass of individual nodes in a material region change so that the node masses are not equal.

Table 3.1		
Heat Transfer Node Characteristics		
<u>Characteristic</u>	<u>Material Nodes</u>	<u>Surface Nodes</u>
node location	mass center	surface
mass	yes (kg)	no
specific heat	yes	no
power	W/kg	W/m ²
phase change	yes	no

The modified Crank-Nicolson finite difference method used for solution of material node temperatures uses a variable forward-backward weighting factor. The forward weighting factor is restricted to the range of 0-0.49999 and the backward weighting factor is 1.0 minus the forward weighting factor. The forward weighting factor is calculated from a linear stability analysis for each material node. The minimum value determined from all material nodes in an axial layer is used in the calculation of temperatures for that layer. The modified Crank-Nicolson formulation is:

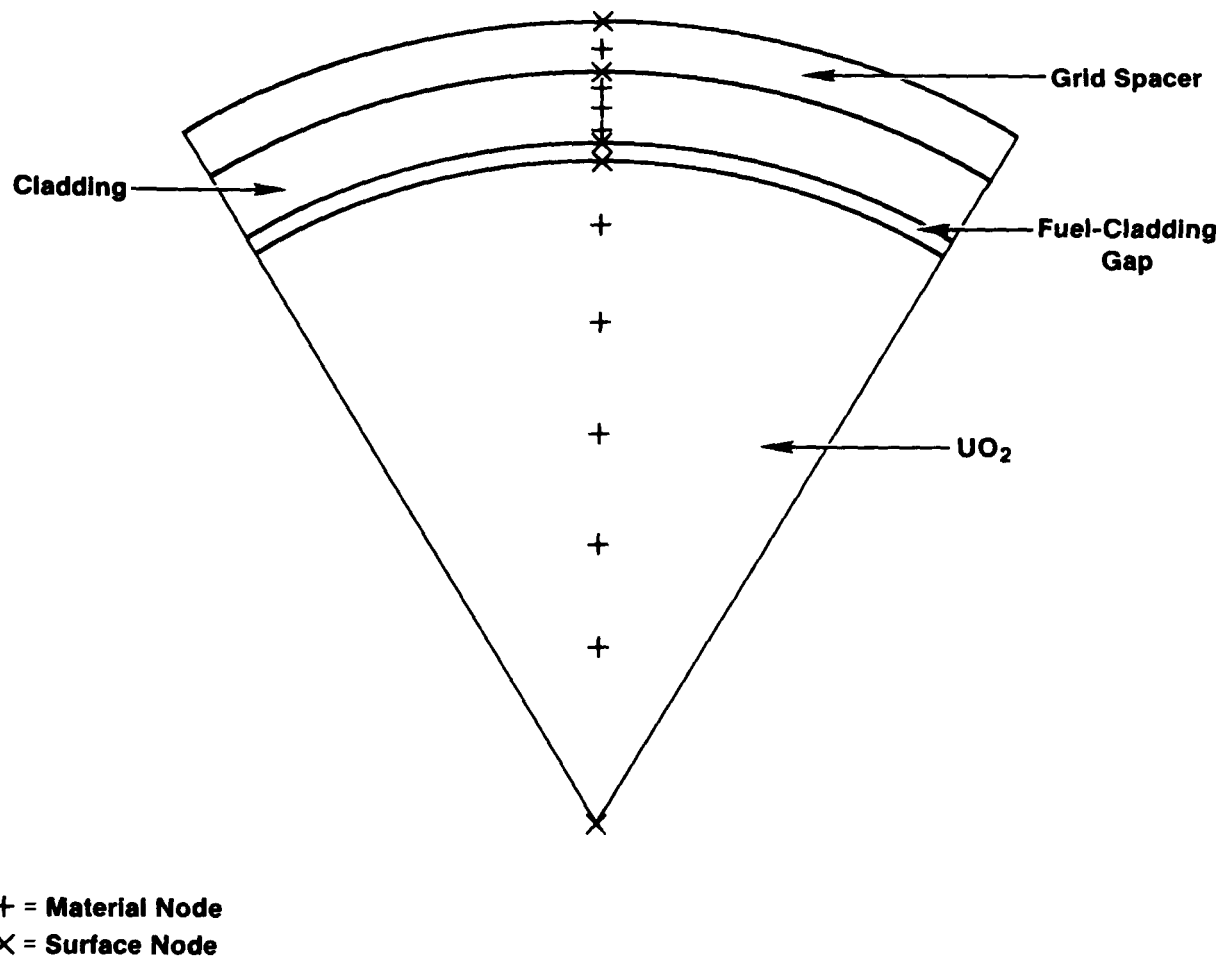


Fig. 3.6 Node arrangement for a fuel rod.

$$\frac{(mC_p)_i^n (T_i^{n+1} - T_i^n)}{\Delta t} = (1 - \beta^n) \sum_{j=1}^{N^n} \left[k_{ij}^n \left(\frac{A}{\Delta x} \right)_{ij}^n (T_j^{n+1} - T_i^{n+1}) \right] + \beta^n \sum_{j=1}^{N^n} \left[k_{ij}^n \left(\frac{A}{\Delta x} \right)_{ij}^n (T_j^n - T_i^n) \right] + m_i^n Q_i^n \quad (3.1)$$

where

- T_i^{n+1} = temperature in node i at time step n+1 (k),
 T_i^n = temperature in node i at time step n (k),
 C_{pi}^n = specific heat of node i at time step n (J/kg),
 m_i^n = mass of node i at time step n (kg),
 Δt = time step (s),
 β^n = forward weighting factor at time step n ($0 \leq \beta^n \leq 0.49999$),
 N^n = total number of temperature nodes at time step n,
 k_{ij}^n = thermal conductivity between nodes i and j and at time step n (W/m K),
 $\left(\frac{A}{\Delta x} \right)_{ij}^n$ = heat transfer geometry factor between nodes i and j at time step n (m), and
 Q_i^n = power per unit mass in node i at time step n (W/Kg).

The forward weighting factor (β), obtained from a linear stability analysis, is calculated as follows:

$$\beta_n = \text{minimum} \left(\beta_1^n, \beta_2^n, \dots, \beta_i^n, \dots, \beta_N^n \right). \quad (3.2)$$

where

$$\beta_i^n = \frac{(mC_p)_i^n / \Delta t}{\sum_{j=1}^N k_{ij}^n \left(\frac{A}{\Delta x} \right)_{ij}^n}.$$

Surface nodes are nodes that have no mass and, therefore, no specific heat. For these nodes the temperature calculation must be formulated in terms of the new time step temperatures only. This equation is:

$$0 = \sum_{j=1}^{N^n} \left[k_{ij}^n \left(\frac{A}{\Delta x} \right)_{ij}^n (T_j^{n+1} - T_i^{n+1}) \right] + A_i^n Q_i^n \quad (3.3)$$

where the terms are as above except for

Q_i^n = surface heat flux at node i at time step n
(W/m²), and

A_i^n = surface area of node i at time step n (m²).

In Eqs. (3.1) and (3.2) the thermal conductivities k_{ij}^n are calculated by averaging the thermal conductivity at nodes i and j, at time step n, that is:

$$k_{ij}^n = \frac{k_i^n + k_j^n}{2} \quad (3.4)$$

Also, in Eqs. (3.1) and (3.3) the geometry factors, for heat transfer between nodes i and j are calculated at each time step and depend on the geometry (cylindrical or slab) and on the node locations (boundary or interior). For interior nodes in cylindrical geometry

$$\left(\frac{A}{\Delta x} \right)_{ij}^n = \left[2\pi \left(\frac{(r_j^n)^2 + (r_i^n)^2}{2} \right)^{1/2} \Delta z^n \right] |r_j^n - r_i^n|^{-1}, \quad (3.5)$$

where

r_i^n = radius of node i at time step n (m),

r_j^n = radius of node j at time step n (m), and

Δz^n = axial height of nodes in the axial layer at time step n (m).

For interior nodes in slab geometry

$$\left(\frac{A}{\Delta x} \right)_{ij}^n = \frac{L^n \Delta z^n}{|r_j^n - r_i^n|} \quad (3.6)$$

where

L^n = structure width at time step n (m),

Δz^n = axial height of nodes in the axial layer at time step n (m),

r_i^n = location of node i at time step n (m), and

r_j^n = location of node j at time step n (m).

Surface nodes on a CORE structure boundary are treated as a special case. (Note that material nodes are never on a boundary.) For these nodes the temperature equation is written as

$$0 = \sum_{k=1}^M \left[H_k^{n+1} A_i^n (T_k^{n+1} - T_i^{n+1}) \right] + \sum_{j=1}^{N^n} \left[K_{ij}^n \left(\frac{A}{\Delta x} \right)_{ij}^n (T_j^{n+1} - T_i^{n+1}) \right] + A_i^n Q_i^n \quad (3.7)$$

where

M = number of convective boundary conditions,

H_k^{n+1} = heat transfer coefficient for the kth boundary condition at time step n+1 (W/m^2K),

A_i^n = area of surface i at time step n (m^2),

T_k^{n+1} = temperature of fluid component k at time step n+1 (K),

T_i^{n+1} = temperature of surface node i at time step n+1 (K),

N^n = number of temperature nodes,

k_{ij}^n = thermal conductivity between nodes i and j at time step n ($W/m K$),

$\left(\frac{A}{\Delta x} \right)_{ij}^n$ = heat transfer geometry factor between nodes i and j at time step n (m),

T_j^{n+1} = temperature of node j at time step n+1 (K), and

Q_i^n = surface power source for node i at time step n (W/m^2).

Thermal conductivity and geometry factors used in Eq.(3.7) are calculated in the same manner for boundary surface nodes as for interior nodes, Eq.(3.4) and Eqs.(3.5) and (3.6), respectively. Heat transfer coefficients and fluid component temperatures are transferred from the FLUIDS module to the CORE module. The surface power source consists of a thermal radiation flux and, depending on conditions, can include oxidation energy and energy

from fission product plateout. A detailed discussion of oxidation kinetics is given in Section 3.3. Thermal radiation fluxes are calculated in the RADIATION module while fission product plateout energy sources are calculated in the VICTORIA fission product module.

3.2.2.2 Phase Transition Heat Transfer

Phase transitions are treated as a bulk phenomenon except for conduction heat transfer limited crusting, which is discussed in the next section. Phase transitions include melting, freezing, and solid-solid transitions. The solution procedure involves an iteration on the finite difference temperature solution. All phase transitions are assumed to occur over a temperature range of 1.0 K or more. During the iteration an "effective" specific heat is used in the temperature calculation. The "effective" specific heat is a combination of the phase transition energy and the actual specific heat.

The phase transition iteration loop consists of three sections. The first part is for phase transitions with decreasing temperatures. The second part is for phase transitions with increasing temperatures. And the third section is for both types of transitions. In parts one and two the net energy change in each node undergoing a phase transition and a corrected temperature are calculated. For increasing temperature conditions the net energy change is

$$Q_i = C_{p_{eff_i}} (T_i^{n+1} - T_i^n) \quad (3.8)$$

where

Q_i = net energy change in node i (J),

$C_{p_{eff_i}}$ = "effective" specific heat of node i (J/K)

$T_i^{n+1,k}$ = temperature of node i at time step n+1 and iteration k (K), and

T_i^n = temperature of node i at time step n (K).

The adjusted temperature is calculated as follows. First, if T_i^n is less than the threshold temperature for the phase transition the value of Q_i is adjusted for the energy needed to raise the temperature to the threshold. That is, Q_i' is given by:

$$Q'_i = Q_i + C_{p_i} (T_i^n - T_{l_i}) \quad (3.9)$$

where

C_{p_i} = true specific heat of node i (J/K) and

T_{l_i} = threshold temperature for the phase transition (K).

The adjusted temperature is then calculated as

$$T_i^{n+1,k'} = Q'_i / SC + \text{maximum} (T_i^n, T_{l_i}) \quad (3.10)$$

where

$$SC = C_{p_i} + H_{\text{phase}_i} \text{ and}$$

H_{phase_i} = phase change energy expressed in units of J/K.

If $T_i^{n+1,k'}$ is greater than the upper phase transition temperature a further adjustment is necessary. The adjusted temperature is then calculated as follows:

$$Q'_i = Q'_i + SC \left[\text{maximum} (T_i^n, T_{l_i}) - T_{h_i} \right] \quad (3.11)$$

where

T_{h_i} = upper phase transition temperature for node i.

and

$$T_i^{n+1,k'} = Q'_i / C_{p_i} + T_{h_i} .$$

In parts one and two there is also a second procedure used to calculate the energy change in each node undergoing a phase transition. This procedure assumes that the temperature $T_i^{n+1,k}$ is correct and that the energy change in the node must be adjusted.

The adjusted energy for conditions in which the node temperature is increasing is calculated as:

$$Q_p = C_{p_i} \left(T_i^{n+1,k} - T_i^n \right) + H_{\text{phase}_i} \left[\text{maximum} \left(T_{l_i}, \text{minimum} \left(T_i^{n+1,k}, T_{h_i} \right) \right) - \text{minimum} \left(T_{h_i}, \text{maximum} \left(T_i^n, T_{l_i} \right) \right) \right] \quad (3.12)$$

Completely analogous procedures are used to calculate $T_i^{n+1,k'}$ and Q_{p_i} for conditions in which the node temperature is decreasing. In part three these results are used to calculate two new "effective" specific heats as follows:

$$C_{p_{\text{eff}_{i,1}}} = Q_{p_i} / \left(T_i^{n+1,k} - T_i^n \right) \quad (3.13a)$$

and

$$C_{p_{\text{eff}_{i,2}}} = Q_i / \left(T_i^{n+1,k'} - T_i^n \right) \quad (3.13b)$$

If $T_i^{n+1,k'}$ is less than T_{l_i} or $T_i^{n+1,k}$ is greater than T_{h_i} , then $C_{p_{\text{eff}_{i,1}}}$ is used in the next phase transition iteration otherwise $C_{p_{\text{eff}_{i,2}}}$ is used. If convergence is not achieved, a relaxation procedure is used to continue the iteration.

The relaxation procedure is triggered if convergence is not reached in MRELAX iterations (currently 15). This procedure constrains $C_{p_{\text{eff}_i}}^{k+1}$ to

$$C_{p_{\text{eff}_i}}^{k+1} = \text{minimum} \left[\text{CHANGP} * C_{p_{\text{eff}_i}}^k, \text{maximum} \left(\text{CHANGM} * C_{p_{\text{eff}_i}}^k, \frac{C_{p_{\text{eff}_{i,1}}} + C_{p_{\text{eff}_{i,2}}}}{2.0} \right) \right] \quad (3.14)$$

where

CHANGP = 1.0 + RELAXO initially

and

CHANGM = 1.0 - RELAXO initially.

$C_{p_{\text{eff}}}^{k+1}$ is also constrained to the range between $C_{p_{\text{eff}_{i,1}}}$ and $C_{p_{\text{eff}_{i,2}}}$. In every NRELAX iteration (currently 3), the values of CHANGP and

CHANGM are recalculated as

$$\text{CHANGP} = (\text{CHANGP} - 1.0) * \text{RELAXN} + 1.0$$

and

$$\text{CHANGM} = 1.0 - (1.0 - \text{CHANGM}) * \text{RELAXN}.$$

The values of CHANGP and CHANGM are adjusted until CHANGP-1.0 is less than PHCNVG*RELAXM. PHCNVG is the phase change convergence criteria and RELAXM is a factor (currently set to 5.0).

3.2.2.3 Conduction Limited Freezing

As materials flow (or candle) down a structure from hotter to colder regions, they may freeze and form a crust on the surface. The rate of such freezing is often limited by conduction heat transfer (in the radial direction) from the melt to the structure. This section describes the conduction limited freezing model employed in the analysis. The motion of molten materials is calculated in the CANDLE subroutine that is discussed in Section 3.2.6.

The model treats a flowing film at temperature T_2 on a solid surface at temperature T_1 . The interface temperature at the surface of the solid is T_s . The distance from T_1 to T_s is Δx_1 and the distance from T_s to T_2 is Δx_2 . (This arrangement is shown in Figure 3.7.) The energy transfer rate at T_s is then defined to be

$$\frac{k_{1,s} A_{1,s}}{\Delta x_1} (T_s - T_1) = \frac{k_{s,2} A_{s,2}}{\Delta x_2} (T_2 - T_s) + Q_s A_s \quad (3.15)$$

where

$k_{c,m}$ = thermal conductivity of crust material between T_c and T_m (W/mK),

$A_{c,m}$ = heat transfer area between T_c and T_m (m^2),

δ_c = growth of crust between time t and $t + \Delta t$ (m),

T_c = temperature of crust (K),

$k_{m,2}$ = thermal conductivity of melt between T_m and T_2
(W/mK),

L_c = heat of fusion of crust (J/kg),

ρ_c = density of crust (kg/m³),

δ = rate of crust growth (m/s),

A_c = surface area of the crust (m²),

$\Delta x_2' = \Delta x_2 - \delta_c$, and

all other variables are as previously defined.

For a crust to form T_s must be less than the freezing point T_m of the liquid film. Also, T_s must be greater than T_1 for heat to flow into the solid. If $T_s - T_1 > 0.0$ and $T_s < T_m$ then a crust of thickness δ can grow in time step Δt . The interface temperature between the melt and the crust must be T_m . (If the crust surface temperature drops below T_m then all of the melt has frozen.) Figure 3.8 shows the arrangement of temperatures after a crust has started to form.

The amount of crusting that can occur in time step Δt is determined by the integral of the energy transfer into the solid layer. At time t the heat flux is given by Eq.(3.15) while at time $t + \Delta t$ the thickness of the crust must be accounted for. At time $t + \Delta t$ the heat flowing into the crust region is given by

$$\frac{k_{c,m} A_{c,m}}{\delta_c/2} (T_m - T_c) = \frac{k_{m,2} A_{m,2}}{\Delta x_2'} (T_2 - T_m) + L_c \rho_c \delta A_c \quad (3.16)$$

Equation (3.16) can be simplified if the heat capacity of the growing crust is not treated directly. Then the temperature T_c does not have to be calculated because the heat flux at T_m must equal the heat flux at T_s and Eq.(3.16) becomes

$$\frac{k_{c,s,m} A_{s,m}}{\delta_c} (T_m - T_s) = \frac{k_{m,2} A_{m,2}}{\Delta x_2'} (T_2 - T_m) + L_c \rho_c \delta A_c \quad (3.17)$$

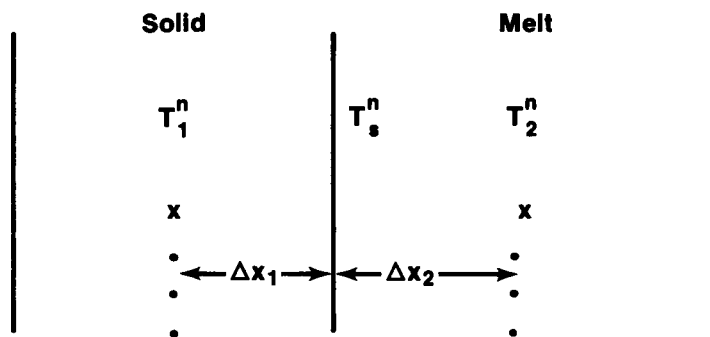


Fig. 3.7 Geometry at time f , prior to the formation of a crust.

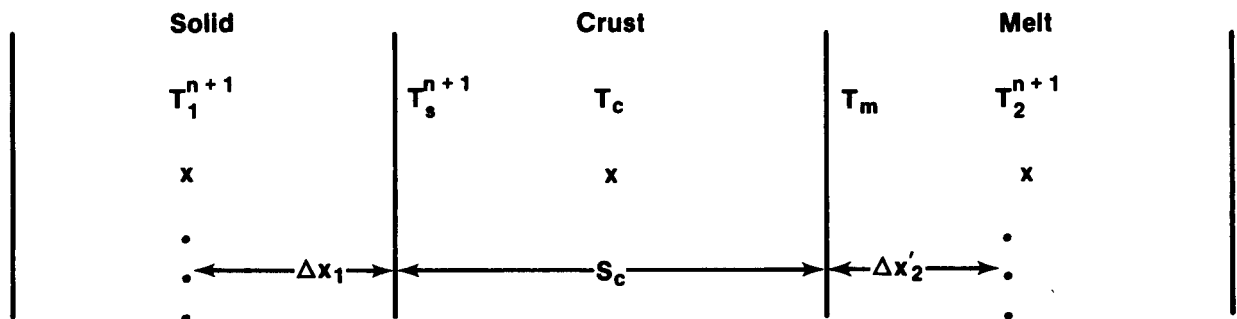


Fig. 3.8 Geometry at time $t + \Delta t$.

where

k_c = thermal conductivity of the crust (W/mK),

$A_{s,m}$ = heat transfer area between T_s and T_m (m^2),

and all other variables are as previously defined.

Equation (3.17) can be simplified further because the energy flow through the crust, together with the source term at the crust solid interface, must equal the energy flowing into the solid, that is:

$$\frac{k_{1,s} A_{1,s}}{\Delta x_1} (T_s - T_1) = \frac{k_c A_{s,m}}{\delta_c} (T_m - T_s) + Q_s A_s \quad (3.18)$$

which yields

$$\frac{k_{1,s} A_{1,s}}{\Delta x_1} (T_s - T_1) = \frac{k_{m,2} A_{m,2}}{\Delta x_2'} (T_2 - T_m) + L_c \rho_c \dot{\delta} A_c + Q_s A_s \quad (3.19)$$

where all terms are as previously defined.

The heat of fusion energy associated with crusting is in reality deposited at the point where freezing is taking place. However, because in this model the heat capacity of the crust is not treated directly, the heat of fusion energy can be deposited at the solid-crust interface without any further loss in accuracy. Because the heat of fusion energy is deposited at the solid-crust interface the only change required to the standard heat transfer solution equations is the addition of a source term at T_s .

To calculate the integral of the energy conducted into the solid during time Δt the average of the heat transfer rates at times t and $t+\Delta t$ is used (Eqs. (3.15) and (3.19)). The resulting equation is

$$\begin{aligned} & \frac{k_{1,s}^n A_{1,s}^n}{2\Delta x_1^n} (T_s^n - T_1^n) + \frac{k_{1,s}^{n+1} A_{1,s}^{n+1}}{2\Delta x_1^{n+1}} (T_s^{n+1} - T_1^{n+1}) \\ &= \frac{k_{s,2}^n A_{s,2}^n}{2\Delta x_2^n} (T_2^n - T_s^n) + \frac{k_{n,2}^{n+1} A_{m,2}^{n+1}}{2\Delta x_2^{n+1}} (T_2^{n+1} - T_m) \\ &+ L_c \rho_c \dot{\delta} A_c + Q_s A_s \end{aligned} \quad (3.20)$$

The geometry between T_1 and T_s does not change over the time step and the thermal conductivity between T_1 and T_s is assumed constant over the time step. Therefore,

$$k_{1,s}^n = k_{1,s}^{n+1} ,$$

$$A_{1,s}^n = A_{1,s}^{n+1} ,$$

and

$$\Delta x_1^n = \Delta x_1^{n+1} .$$

When the above equalities are substituted into Eq.(3.20), the following result is obtained.

$$\begin{aligned} & \frac{k_{1,s} A_{1,s}}{2\Delta x_1} [T_s^n - T_1^n - T_s^{n+1} - T_1^{n+1}] \\ &= \frac{k_{s,2}^n A_{s,2}^n}{2\Delta x_2^n} (T_2^n + T_s^n) + \frac{k_{m,2}^{n+1} A_{m,2}^{n+1}}{2\Delta x_2^{n+1}} (T_2^{n+1} - T_m) \\ & \quad + L_c \rho_c \dot{\delta A}_c + Q_s A_s . \end{aligned} \quad (3.21)$$

Equation (3.21) is not quite correct because from the instant the crust begins to form the temperature at the crust-melt interface is T_m . Therefore, T_s^n on the right hand side of Eq(3.21) should be T_m . Also, if

$$L_c \rho_c \dot{\delta A}_c = L_c \dot{m} ,$$

$$\frac{k_{1,s} A_{1,s}}{2\Delta x_1} = G_1^n ,$$

$$\frac{k_{s,2}^n A_{s,2}^n}{2\Delta x_2^n} = G_2^n ,$$

and

$$\frac{k_{m,2}^{n+1} A_{m,2}^{n+1}}{2\Delta x_2^{n+1}} = G_2^{n+1}$$

then Eq.(3.21) can be written as

$$G_1^n [T_s^n - T_1^n + T_s^{n+1} - T_1^{n+1}] \\ = G_2^n (T_2^n - T_m) + G_2^{n+1} (T_2^{n+1} - T_m) + L_c \dot{m} + Q_s A_s. \quad (3.22)$$

As stated above, the heat capacity of the growing crust is not treated directly. However, it is treated by modifying the definition of L_c to include a specific heat term. L_c is then defined as

$$L_c = \text{MIN}[H_F, H_2 - H_m] + C_p [T_2^{n+1} + T_2^n - (T_s^{n+1} + T_s^n)/2 - T_m] \quad (3.23)$$

where

H_F = heat of fusion of the crust (J/kg),

H_2 = enthalpy of melt (J/kg),

H_m = enthalpy of melt at solids (J/kg),

and

C_p = specific heat of crust (J/kgK).

In Eq.(3.23), the energy associated with the specific heat of the crust is based on a linear temperature profile through the crust. The true temperature profile is not linear, however; the linear assumption doesn't introduce much error because $cp \ll H_F$.

In Eq.(3.22) there are six unknowns: T_1^{n+1} , T_2^{n+1} , G_2^{n+1} , T_s^{n+1} , L_c , and $\dot{m}$. Of these, T_1^{n+1} , T_2^{n+1} , and T_s^{n+1} are provided by the standard temperature solution which is performed iteratively when crusting is occurring. If C_p is treated as a constant, then L_c is only a function of temperatures T_1^{n+1} , T_2^{n+1} , and T_s^{n+1} . These temperature are calculated at each iteration. At time t

$$G_2^{n+1} = G_2^n$$

and at time $t+\Delta t$

$$G_2^{n+1} = f(G_2^n, \dot{m}\Delta t).$$

In slab geometry

$$G_2^{n+1} = G_2^n \left(\frac{m_2}{m_2 - \dot{m}\Delta t} \right).$$

In cylindrical geometry this simple ratio is not correct; however, for $\Delta r/r \ll 1$ a slab approximation produces only a slight error. G^{n+1} is calculated during the iteration and is a function of the previous iterate value of $\dot{m}\Delta t$.

During each iteration $\dot{m}\Delta t$ is solved from Eq.(3.22) as

$$\dot{m}\Delta t = \Delta t \left[G_1^n (T_s^n - T_1^n + T_s^{n+1} - T_1^{n+1}) - Q_s A_s - G_2^n (T_2^n - T_m) - G_2^{n+1} (T_2^{n+1} - T_m) \right] / L_c \quad (3.24)$$

This iteration continues until the relative change in $\dot{m}\Delta t$ is less than a user input value or until the iteration limit is reached.

Conduction limited crusting is calculated when the bulk enthalpy of the melt remains above the solidus enthalpy over the time step Δt . If the phase change calculation determines that the bulk temperature of the melt drops below the solidus, then only bulk freezing is treated. The model has undergone limited testing, therefore, an evaluation of optimum axial mesh size and appropriate time steps has not been determined.

3.2.3 Chemistry

Two types of chemical interactions are considered in the analysis: oxidation and eutectic formation. The zirconium and stainless steel oxidation models are described in the next section. Eutectic formation and the dissolution of UO_2 and Zr_2 by molten Zr are discussed in Section 3.2.3.2.

3.2.3.1 Oxidation Reactions

The zirconium and steel oxidation kinetics implemented in MELPROG are based on correlations developed by Prater and Courtright [3.2] and White, et al. [3.3], respectively. In both cases the oxidation rates depend only on temperature as long as both steam and metal are both available. After a thin oxide layer is formed, the oxidation rates for zirconium and steel are of the same order of magnitude at a given temperature. However, the energy released per mass of oxygen consumed is an order of magnitude greater for zirconium than for steel.

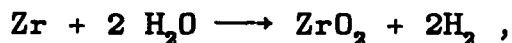
When significant amounts (>1%) of the zircaloy-4 cladding are oxidized, the CORE module uses several regions to model heat transfer in the cladding. Typically, the model calculates the growth of a ZrO_2 layer on the outside of the cladding and a α -Zr layer between the Zr and ZrO_2 regions. For simplicity it is assumed that the α -Zr has a constant oxygen concentration of 20 atom percent. The rates at which these layers grow is limited by the rate at which oxygen diffuses through the oxide and the thickness L [m] of these layers (ZrO_2 and α -Zr+ ZrO_2) is given by:

$$\frac{dL}{dt} = \delta^2 \quad , \quad (3.25)$$

where the growth rate δ is given by

$$\delta = A e^{-B/T} \quad . \quad (3.26)$$

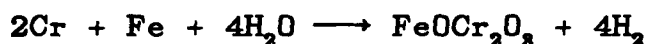
The constants A and B are given in Table 3.2. Note that the reaction rates increase dramatically at a temperature of 1820 K. At this temperature the ZrO_2 lattice changes from a tetragonal to a cubic structure and the oxygen diffusivity increases significantly. For the oxidation of U-Zr-O mixtures Prater and Courtright give for A and B: $A=0.0748$ and $B=14780.0$. The complete reaction of Zirconium with steam is:



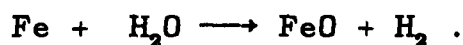
and the heat of reaction is on the order of 6.45×10^8 J/kg of Zr reacted.

Table 3.2			
Parameters for Parabolic Kinetics Equations (Zr oxidation)			
<u>Layer</u>	<u>Temperature Range K</u>	<u>A [m/s^{1/2}]</u>	<u>B [K]</u>
ZrO ₂	1000-1820	0.0100	12230.0
	1820-2670	0.0546	14241.0
α -Zr + ZrO ₂	1000-1820	0.0006	6160.0
	1820-2170	0.0205	11550.0

The reaction of stainless steel in a steam environment is assumed to be a combination of:



and



The heat of reaction used is 8.7×10^5 J/kg of steel reacted. An examination of the equilibrium data for stainless steel constituents indicates that the first reaction should take place even in the presence of substantial quantities of hydrogen (up to 97%), while the second reaction will occur while no more than half the mixture is hydrogen.

The growth rates for steel were taken from the experiments of J.F. White for 304L stainless steel and should be applicable for all steels with approximately 19% Cr content. In his analysis, linear and parabolic models for the oxygen consumption rates were determined in the form of:

$$\delta = A e^{-B/T} , \quad (3.27)$$

where the coefficients are shown in Table 3.3. He determined that the linear behavior was due to iron oxide formation on the outer surface and the parabolic was due to the chromium oxide formation further in. In our formulation we determine the oxygen uptake of the oxide layer at which the parabolic behavior will dominate and then by knowing the current oxygen weight gain we choose the dominate oxygen uptake mechanism. In addition there is a correction for the linear behavior in the presence of excess hydrogen.

Table 3.3		
Parameters for Steel Kinetics Equations (Steel Oxidation)		
	<u>A</u>	<u>B</u>
linear behavior	1100.0 kg/m ² s	22308.0 K
parabolic behavior	2.40×10^8 kg ² /m ⁴ s	42401.5 K

3.2.3.2 Eutectic Formation

The CREUTK subroutine uses a simple algorithm to determine how much solid material can be dissolved in a given quantity of molten material of a given composition and temperature. The routine was initially devised to treat UO_2 dissolution by molten U-Zr-O eutectic, but the algorithm is general and not limited to those materials. The routine can treat the dissolution of two different solids on opposite sides of the molten material (e.g., UO_2 pellet and ZrO_2 shell separated by molten U-Zr-O); however, because the routine uses an equilibrium model with no dissolution rate information and treats the two solids consecutively rather than simultaneously, the comparative rates of dissolution of the two solids may be incorrect. An exact treatment of simultaneous dissolution is not feasible at this time.

Figure 3.9 is a flow chart of the CREUTK subroutine. Basically, the routine determines how much solid may be added to the mixture before the resulting specific enthalpy equals the liquidus enthalpy of the new composition. The solution algorithm is iterative and unconditionally stable. MCOMP(K) contains the initial masses of the various components that define the mixture, and ISOL is the value of K that refers to the solid component that is being dissolved. The CX(K) array is equal to the MCOMP(K) array plus the mass of new dissolved solid added to MCOMP(ISOL). CREUTK calls MELPRO to determine the initial solid enthalpy and the liquidus enthalpy of the mixture as a function of its changing composition between iterations. The DELM(ISIDE) and DELE(ISIDE) terms are the mass and energy, respectively, of the solid on either side that is dissolved during the time step. The algorithm allows solid dissolution at an interface where the mixture is oxidizing as long as the rate of dissolution of the ZrO_2 is greater than its rate of formation by oxidation. This requirement prevents the simultaneous creation of a new oxide layer and dissolved solid region at the same interface.

3.2.4 Energy Generation

The CORE module considers energy generation due to nuclear fission and oxidation. Power factors are specified in the input (see Appendix A) for each material region, axial cell and radial ring. These factors are normalized by the code such that

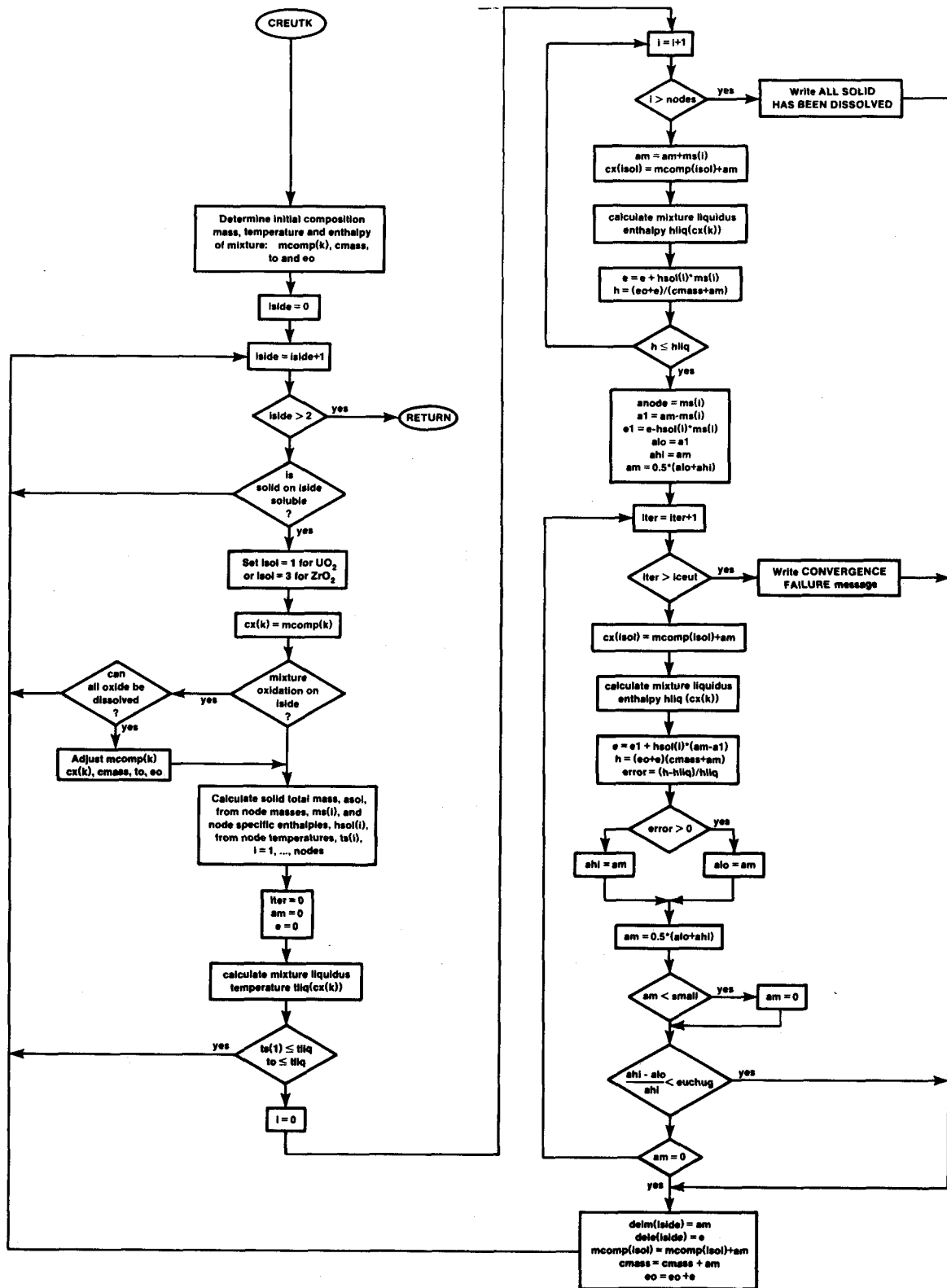


Fig. 3.9 Flow chart of the logic in the eutectic formation subroutine, CREUTK.

$$\frac{\sum_i m_i P_i P_z P_r}{\sum_i m_i} = 1 \quad , \quad (3.28)$$

where

m_i = mass of the material region,

P_i = power factor associated with material region i ,

P_z = power factor associated with the axial cell that contains region i , and

P_r = power factor associated with the radial ring that contains region i .

At the beginning of a calculation, the total reactor power Q_{tot} and a time-dependent decay function $f(t)$ must also be specified. The energy generation in a given material region is then $Q_{tot} f(t) P_i P_z P_r m_i$.

While fission decay heat is treated on a volumetric basis, energy generated by oxidation is deposited at an interface. For example, the energy released by Zr oxidation (see Section 3.2.3.1) is deposited at a surface of the ZrO_2 layer.

3.2.5 Fuel Rod Mechanics

The structure of the CORE module is sufficiently flexible that a wide range of mechanical phenomena can be treated. For example, the heat transfer calculation can accommodate the change in fuel rod diameter associated with thermal expansion, ballooning and swelling. The candling model that will be discussed in Section 3.2.6 allows melt to flow through holes that can form in the cladding. In the current version of the code, it is assumed that the cladding fails when the temperature reaches a user specified value. In the near future, correlational models will be added to the CORE module to determine if and when the fuel rods balloon, swell or rupture.

3.2.6 Candling

The CANDLE subroutine determines how the molten material moves when core structures begin to melt. Motion of the molten core material is important because it strongly influences several phenomena that affect the severity of LWR accident sequences. For example, candling contributes to both the formation and elimination of flow blockages, it affects oxidation by either increasing or decreasing metal supplies and steam access to those

supplies, and it can accelerate or inhibit material interactions such as eutectic formation. CANDLE is called by the CORE subroutine whenever any part of a CORE structure model is found to be molten (MDRMSH=2). CORE provides CANDLE with model data and the starting address (NCNT2) of all the other sub-model data that is required to define and solve the candling problem. CANDLE consists of two parts. In the first part, the routine loops through the axial layers of the core structure model to determine the geometry of the flow paths and the location and initial conditions of molten material. In the second part, mass, energy and momentum conservation equations are solved along the axial flow paths to determine the extent of motion during the time step. Each part will be discussed in more detail in the following sections.

3.2.6.1 Problem Definition

CANDLE begins defining the problem by determining which fluid cell contains the bottom of the core structure. With this reference point, the appropriate fluid pressure for each axial layer in the core model can be determined. In the initialization loop CANDLE loops through the axial layers starting from the bottom. At each layer CANDLE loops through radial regions that define the geometry and composition of the layer. CANDLE searches for existing candling regions and vacant regions into which molten material may flow.

There are two kinds of candling regions. Type 5 candling regions (MDTYPR=5) occur between two bounding solid regions such as fuel and cladding and are characterized by homogeneous flow assumptions, while type 6 candling regions (MDTYPR=6) occur on the surface of solids such as the outer surface of cladding and are characterized by film flow assumptions. When candling or flow regions are identified, the necessary region data is transferred to working arrays that include liquid mass and composition (LCMLIQ and LCCOMP), solid mass (LCMSOL), liquid enthalpy (LCELIQ), liquid velocity (LCVLIQ), region pressure (LCPRGN) and geometric detail (LCAREA, LCVOLF and LCDZ). Then CANDLE determines which candling regions from the previous layer are connected to the newly identified region (LCTJCT and LCBJCT arrays). Candling regions are connected only if they overlap by at least the half thickness of the thinner region or a specified minimum thickness, GAPMIN, whichever is less.

In addition to candling (including flow) regions, each layer

is searched for porous solid regions between candling regions. CANDLE permits flow between candling regions in the same layer that are separated by porous regions such as cracked cladding or guide tubes fabricated with holes. This information is stored in the LCCRAK array, which specifies the candling regions that are involved and the flow resistances.

After looping through all of the radial regions in a layer, CANDLE checks the LCTJCT and LCBJCT arrays to identify junctions where more than two candling regions are connected. This information is stored in the LCIJCT array. After looping through all of the axial layers, CANDLE examines the LCIJCT array to construct the LCPATH array, which defines the sequence of candling regions in each distinct "path" that joins the first and last axial layer. Paths that are connected by either shared junctions or porous separations are contained in the same "group."

3.2.6.2 Solution

In part two of the CANDLE subroutine mass, energy and momentum equations are solved iteratively along each flow path in a group of connected paths to determine how far molten material moves during a time step, DT, which is initially set equal to the MELPROG time step, DELTO. The conservation equations are pseudo-two-dimensional--variations in the axial direction are treated simultaneously during each iteration by differencing between axial layers, while flow between connected paths is treated explicitly during each iteration but is updated between iterations. The axial difference equations are solved with a direct tri-diagonal matrix algorithm to obtain "new" iterate values of the dependent variables from "old" ones. Initial conditions are obtained from the arrays constructed in the first part of CANDLE.

The iteration loop continues until either the convergence error between old and new values is less than an input convergence limit, CDCNVG, or until an input iteration limit, ICANDL, is exceeded. If convergence is obtained or the convergence error is not too large after ICANDL iterations, CANDLE updates the dependent variables and checks to determine if the end of the MELPROG time step has been reached. If the convergence error is still too large after ICANDL iterations, CANDLE cuts its time step in half and repeats the process until

Table 3.4		
CANDLE Working Array Descriptions		
Note: CR = candling region ICR = CR identification number (1≤ICR≤MXCNDL) ILAYER = layer identification number (1≤ILAYER≤MXLYRS) NCOMP = species identification number (1≤NCOMP≤MXSPEC+1=MXSPP1) JCT = junction identification number (1≤JCT≤MAXJCT) IPATH = path identification number (1≤IPATH≤MXPATH)		
<u>Pointer</u>	<u>Array Description (units)</u>	<u>Dimension</u> MXLYRS
LCDZ	axial layer thickness (m)	
LCAREA	CR flow area (m**2)	MXLYRS*MXCNDL
LCVOLF	CR flow volume (m**3)	MXLYRS*MXCNDL
LCPRGN	CR pressure (Pa)	MXLYRS*MXCNDL
LCCOMP	molten species masses in CR (kg)	MXLYRS*MXCNDL
LCMLIQ	total molten mass in CR (kg)	MXLYRS*MXCNDL
LCMSOL	total solid mass in CR (kg)	MXLYRS*MXCNDL
LCELIQ	enthalpy of melt in CR (j)	MXLYRS*MXCNDL
LCVLIQ	velocity of melt in CR (m/s)	MXLYRS*MXCNDL
LCALPL	volume fraction of melt in CR (-)	MXLYRS*MXCNDL
LCMO	initial conditions for species masses in solution algorithm (kg)	MXLYRS*MXCNDL *MXSPP1
LCMOLD	old iteration values of species masses in solution algorithm (kg)	MXLYRS*MXCNDL *MXSPP1
LCMNEW	new iteration values of species masses in solution algorithm (kg)	MXLYRS*MXCNDL *MXSPP1

Table 3.4 (cont.)		
LCEO	initial condition for enthalpy of melt in solution algorithm (j)	MXLYRS*MXCNDL
LCEOLD	old iteration value of enthalpy of melt in solution algorithm (j)	MXLYRS*MXCNDL
LCENEW	new iteration value of enthalpy of melt in solution algorithm (j)	MXLYRS*MXCNDL
LCVO	initial condition for velocity of melt in solution algorithm (m/s)	MXLYRS*MXCNDL
LCVOLD	old iteration value of velocity of melt in solution algorithm (m/s)	MXLYRS*MXCNDL
LCVNEW	new iteration value of velocity of melt in solution algorithm (m/s)	MXLYRS*MXCNDL
LCVSAV	old outer iteration value of velocity of melt in solution algorithm (m/s)	MXLYRS*MXCNDL
LCPO	initial condition for pressure in solution algorithm (Pa)	MXLYRS*MXCNDL
LCPOLD	old iteration value of pressure in solution algorithm (Pa)	MXLYRS*MXCNDL
LCPNEW	new iteration value of pressure in solution algorithm (Pa)	MXLYRS*MXCNDL
LCAO	interfacial area between layers along a path (m**2)	MXLYRS
LCMVO	initial condition of product of velocity and melt mass used in momentum equation (kg-m/s)	MXLYRS
LCALPO	initial value of melt volume fraction used in solution algorithm	MXLYRS*MXCNDL
LCDRAG	drag term used in momentum equation(m ⁻¹)	MXLYRS*MXCNDL
LCZMIN	CR low mass cut-off (kg)	MXLYRS*MXCNDL
LCMISC	CR mdtypr indicator-- X(LCMISC)<0, if it is a type 6 CR with no friction; X(LCMISC)>0, if it is a type 6 CR with friction, and the X(LCMISC) is the fixed substrate radius used to calculate film thickness X(LCMISC)=0, if it is a type 5 CR (m)	MXLYRS*MXCNDL

Table 3.4 (cont.)

LCCRAK	"Porous flow" array of the form $AREA_1, ICR_1, AREA_2, ICR_2$ where $area_1$ is the porous flow area in the left side bounding solid, ICR_1 is the CR on the left side of the porous left boundary, $AREA_2$ is the porous flow area in the right side bounding solid and ICR_2 is the CR on the right side of the porous right boundary.	$MXLYRS * MXCNDL * 4$
LCPATH	sequence of icr's defining a distinct path connecting the first and last axial layers; $ICR < 0$ indicates the path branches downward in the layer below, so that the velocity at that interface is multi-valued (one value for each branch).	$MXLYRS * MXPATH$
LCTJCT	upper layer CR connection array is of the form $ICR_{1,1}, ICR_{1,MXCNDL}, ICR_{2,1}, \dots, ICR_{2,MXCNDL}, \dots, ICR_{MXCNDL,1}, \dots, ICR_{MXCNDL,MXCNDL}$ where $ICR_{I,J}$ is the value of ICR of the j^{th} CR from the lower layer that is connected to the $ICR=I$ CR of the upper layer	$MXCNDL ** 2$
LCBJCT	lower layer CR connection array (same as lctjct with "lower and "upper" switched)	$MXCNDL ** 2$
LCIJCT	junction description array-- each description is of the form $ILAYER, ICR_1, \dots, ICR_{MXCNDL+1}$ where $ABS(ILAYER)$ is the upper layer of junction, and $ILAYER < 0$ indicates the junction branches downward and $ICR_1, \dots, ICR_{MXCNDL}$ are the lower CR's and $ICR_{MXCNDL+1}$ is the upper CR, while $ILAYER > 0$ indicates the junction branches upward and ICR_1 is the lower CR and $ICR_2, \dots, ICR_{MXCNDL+1}$ are the upper CR's.	$AXJCT * (MXCNDL + 2)$
LCPJCT	list of junctions and the appropriate branch of the junction from the LCIJCT array associated with a path from the LCPATH array. Of the form $JCT_1, IBRNCH_1, JCT_2,$	$2 * MAXJCT * MXPATH$

Table 3.4 (cont.)		
	IBRNCH ₁ , ... where JCT ₁ , IBRNCH ₁ means that the path contains the JCT ₁ junction described in the LCIJCT array and takes the ABS(IBRNCH ₁)th branch of the branching layer. IBRNCH ₁ <0 if that is the last branch in junction (sign used in path construction logic).	
LCMXO	explicit mass source in the mass conservation equation from porous flow and junction flow (kg/s)	MXLYRS*MXSPP1
LCEXO	explicit enthalpy source in the energy conservation equation from porous flow and junction flow (w)	MXLYRS
LCIMO	implicit mass source in the mass conservation equation from junction flow(s ⁻¹)	MXLYRS
LCMOME	explicit momentum source in the momentum equation from junction flow (nt)	MXLYRS
LCMOMI	implicit momentum source in the momentum equation from porous flow and justification flow (kg/s)	MXLYRS
LCPP	term used in direct tri-diagonal matrix algorithm	MXLYRS
LCQQ	term used in direct tri-diagonal matrix algorithm	MXLYRS*MXSPP1

it succeeds or DT is too small. When the sum of the successful time steps reaches DELTO, CANDLE calculates incremental changes in the dependent variables over that time and passes them to CRMESH. Figures 3.10a and 3.10b are flow charts of the CANDLE solution algorithm, and they should help the reader to understand the following detailed description of the algorithm. Figure 3.10a shows the overall solution algorithm, while Figure 3.10b shows details of the box marked "SOLVE" in Fig. 3.10a. Table 3.4 identifies the variables addressed by the working array pointers in CANDLE.

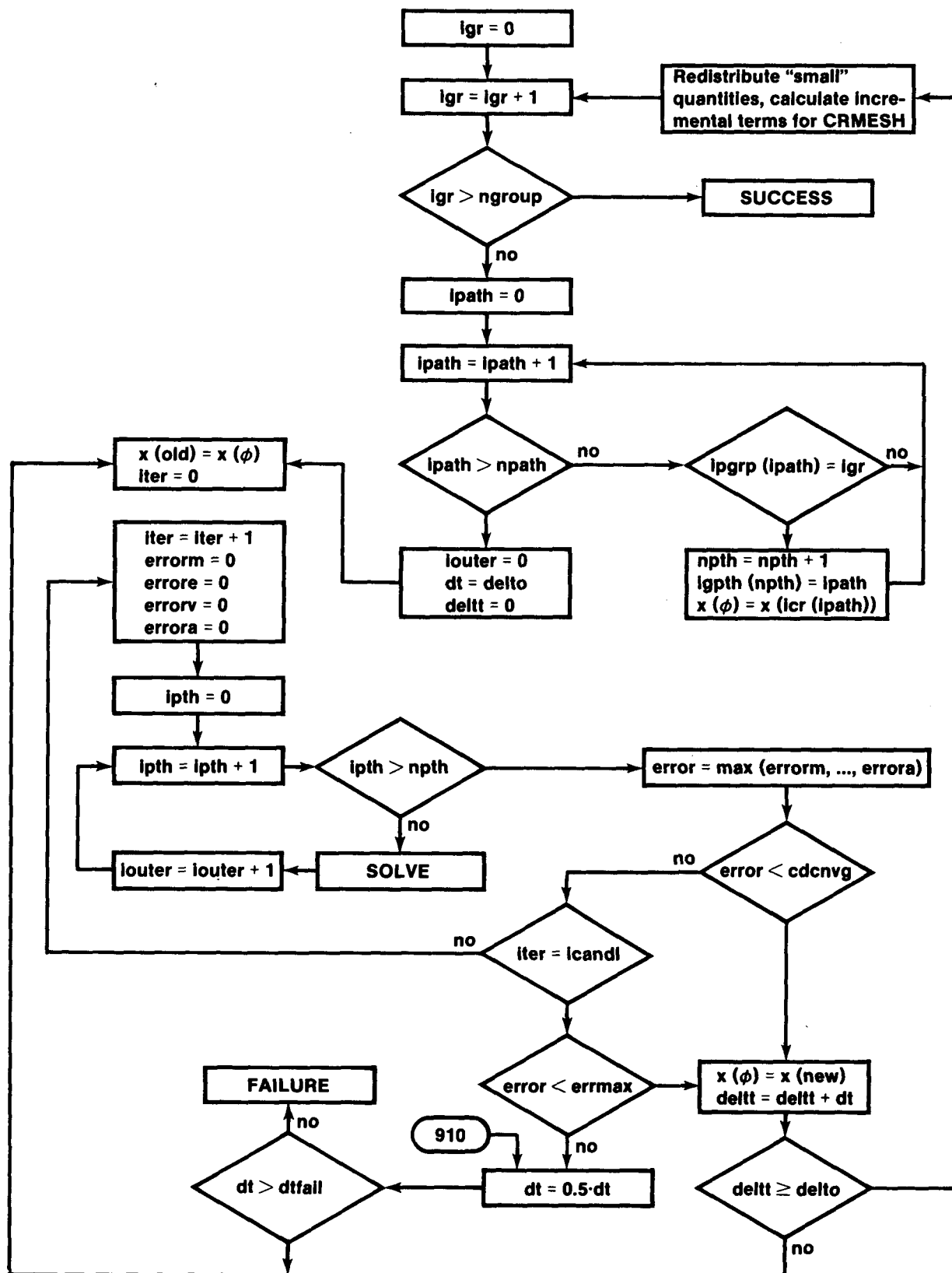


Fig. 3.10a Flow chart showing structure of the CANDLE subroutine.

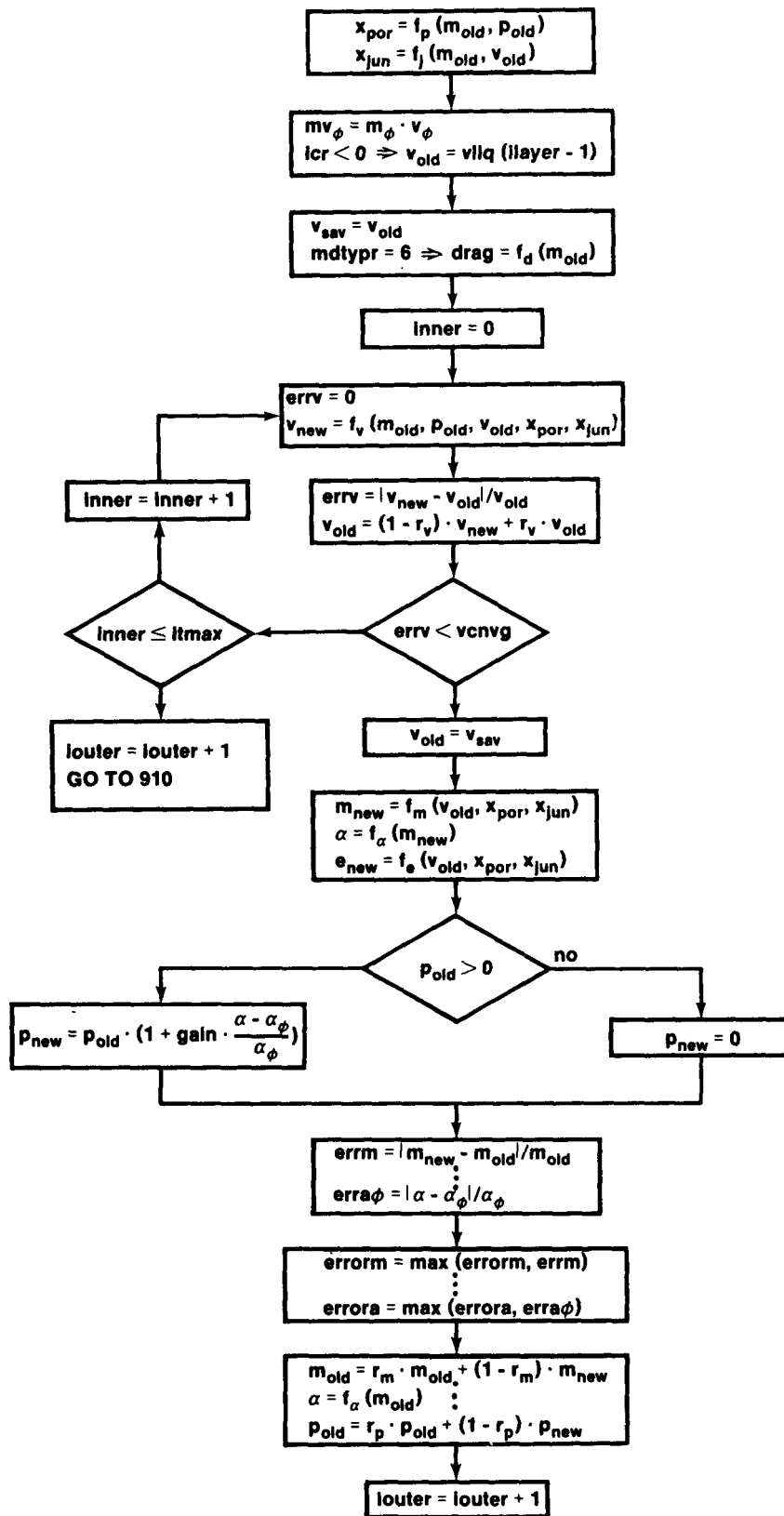


Fig. 3.10b Flow chart showing structure of the CANDLE subroutine. (continued)

3.2.6.3 Iteration Strategy

There are three "layers" to the overall solution algorithm. The outer-most iteration is called the sub-time step iteration and is intended to sub-divide the MELPROG time step when it is necessary to obtain an "acceptable" solution for the conservation equations. Within the sub-time step iteration, the system of conservation equations for all the paths in a group is solved by an outer iteration. And, finally, within each outer iteration the momentum equation along each path in the group is solved by an inner iteration, INNER is the counter for the inner iteration, ITER is the counter for the outer iteration and IOUTER is the total number of outer iterations for all the paths and all the sub-time steps required to solve the system of conservation equations. DT is the current sub-time step, which is initialized to DELTO and DELTT is the current shifted initial time, which is initialized to zero. Each time the outer iteration succeeds DELTT is incremented by DT and the next sub-time step iteration begins. The solution is complete when DELTT equals DELTO.

The subscripts/arguments "0," "old" and "new" in Fig. 3.10a and 3.10b refer to initial values, previous iteration values and current iteration values, respectively. IGR is the group index, NGROUP is the total number of groups, IPATH and IPTH are path indices and NPATH and NPTH are the total number of paths in the model and group, respectively. The "path loop" in Fig. 3.10a determines which paths are included in the current group and loads the initial condition arrays of the dependent variable with the values from Part I.

Each time the conservation equations are solved along any path in the group the path errors ERRM(IPTH), ERRE(IPTH), ERRV(IPTH), and ERRAO(IPTH) are calculated from the old and new value of the mass, energy, velocity, and liquid volume fraction, respectively. The liquid volume fraction error requires some explanation. The pressure is set equal to the region pressure X(LCPRGN) until a candling region is essentially full of molten material. Then, when the initial liquid volume fraction, α_0 or X(LCALPO), exceeds a specified value, α_1 , or ALPHL1, it is assumed that the pressure will rise to whatever value is necessary to prevent further inflow. In fact, the iterative pressure continues to rise as long as the liquid volume fraction at each iteration, α or X(LCALPL), exceeds its initial value α_0 . Therefore, when $\alpha_0 > \alpha_1$, $|\alpha - \alpha_0|/\alpha_0$ gives ERRAO(IPTH), the error associated with over-filling a candling region.

The group errors, ERRORM, ERRORD, etc., are the maximum values of ERRM(IPTH), ERRE(IPTH), etc. over all IPTH, and ERROR is the maximum group error over all dependent variables. After each outer iteration a convergence test is made by comparing the value of ERROR to an input value, CDCNVG. If convergence has not occurred, ITER is compared to an input value, ICANDL. If ITER is less than ICANDL, then it is incremented by one and the next group iteration begins with the old values replaced by the relaxed new values. When ITER reaches ICANDL without convergence, ERROR is compared to a specified acceptance limit, ERRMAX. If ERROR is less than ERRMAX, then the solution proceeds as if convergence had occurred. If ERROR is greater than ERRMAX, then the current sub-time step is halved and the group iteration procedure begins again with the old values replaced by the initial values. If DT should fall below a specified value, DTFAIL, then CANDL fails. If convergence occurs or the error is acceptable, then DELTT is incremented by DT, the initial conditions are updated to the new values of the dependent variables, and the next sub-time step iteration begins.

3.2.6.4 Conservation Equations

Figure 3.10b outlines the solution algorithm for the conservation equations. The X_{por} and X_{jun} terms represent convective terms associated with flow from one path to another either through porous solids separating the paths or from junctions connecting paths. The X_{por} terms are of the form

$$X_w = W_{donor} AV, \quad (3.29)$$

where W_{donor} is the convected quantity (mass, energy or momentum) from the donor cell, and

A = porous area,

$$V = (\Delta P_{old} * d_h / 2f \rho_m \Delta r)^{1/2},$$

and d_h is the hydraulic diameter of the porous flow area, f is the friction factor, ρ_m is the microscopic density of the molten material and Δr is the thickness of the porous solid.

Similarly, the X_{jun} terms are of the form

$$x_w = W_{donor} AV, \quad (3.30)$$

where W is the convected quantity and

A = junction area, and

V = junction velocity.

If the donor cell is outside the path (i.e., either porous flow into the path or flow from a junction into the path) then W is treated explicitly by using the old values of the convected quantity from the donor cell; otherwise, W is treated implicitly within the matrix solution of the conservation equation for the convected quantity (i.e., the new values inside the path are used).

Because the velocities are defined at the bottom of the layers, junctions that branch downward require special treatment. The single cell in the upper layer needs a distinct velocity for flow into each of the lower branches. In this case the value of ICR for the cell in the upper layer is made negative to indicate that the velocity at each interface of the junction is stored in the LCVLIQ location of the branch cell in the lower layer of the junction.

Because the homogeneous flow boundaries of a type 5 candling region are fixed during a time step the geometric drag term is fixed. However, for a type 6 candling region the drag term depends on the film thickness, which changes with the molten mass. Therefore, $X(LCDRAG)$ changes between iterations for type 6 candling regions except for the special case when all the structure in the layer is gone and the drag is set to zero.

3.2.6.5 Momentum Equation

In Fig. 3.10b the momentum equation is designated f_v ; the actual difference equation is:

$$a_i^n v_i^{n+1} = b_i^n v_{i+1}^{n+1} + c_i^n v_{i-1}^{n+1} + d_i^n, \quad (3.31)$$

where

$$\begin{aligned}
a_i^n &= \frac{m_i^n + m_{i-1}^n}{2} + \left[\left(\frac{R_i^O m_i^n}{2} + \frac{R_{i-1}^O m_{i-1}^n}{2} \right) v_i^n + I_{vi}^n \right] \Delta t \\
&+ \frac{m_i^n A_{i+1} \max(0, -v_{i+1/2}^n)}{V_i} \Delta t + \frac{m_{i-1}^n A_{i-1} \max(0, v_{i-1/2}^n)}{V_{i-1}} \Delta t, \\
b_i^n &= \frac{m_i^n A_{i+1} \max(0, v_{i+1/2}^n)}{V_i} \Delta t, \\
c_i^n &= \frac{m_{i-1}^n A_{i-1} \max(0, -v_{i-1/2}^n)}{V_{i-1}} \Delta t, \\
d_i^n &= \frac{m_i^O + m_{i-1}^O}{2} v_i^O + \left[(p_i^n - p_{i-1}^n) A_i + \left(\frac{m_i^n + m_{i-1}^n}{2} \right) g + E_{vi}^n \right] \Delta t, \\
v_{i+1/2}^n &= \frac{v_i^n + v_{i+1}^n}{2}, \\
v_{i-1/2}^n &= \frac{v_i^n + v_{i-1}^n}{2}, \\
R_i &= \frac{2f}{d_{hi}},
\end{aligned}$$

and the symbols are defined in Table 3.5. Because a_i^n , b_i^n and c_i^n depend on v_i^n , the momentum equation is solved iteratively in the inner loop. At the beginning of the inner loop the old velocities, LCVOLD, are copied into the LCVSAV array and the drag terms, LCDRAG, for type 6 candling regions are calculated from the old mass values, LCMOLD. During each iteration Eq.(3.31) is solved using a tri-diagonal matrix algorithm [3.1]. The convergence criterion for the inner loop, VCNVG, changes as the value of ITER (the outer iteration counter) increases. Initially, if there is no volume fraction error along the path, the value of VCNVG is set to 10^{36} ; otherwise, it is set to 10^{-2} . However, after the first outer iteration, VCNVG is set to the minimum of VCNVG and ERRX, where ERRX is the maximum of ERRM(IPTH), ERRE(IPTH) and CDCNVG. If the inner iteration has not converged after ITMAX iterations, the time step is halved and the outer iteration loop is repeated. When the inner iteration converges, the old velocities are re-set with the values saved in the LCVSAV array, and the outer iteration moves to the mass and energy equations.

Table 3.5

Conservation Equation Symbols Used in the CANDLE Subroutine

<u>Symbol</u>	<u>Description</u>	<u>Pointer</u>
m_i^n	old mass at layer i	LCMOLD
m_i^{n+1}	new mass at layer i	LCMNEW
e_i^n	old enthalpy at layer i	LCEOLD
e_i^{n+1}	new enthalpy at layer i	LCENEW
v_i^n	old velocity at interface i	LCVOLD
v_i^{n+1}	new velocity at interface i	LCVNEW
p_i^n	old pressure at layer i	LCPOLD
p_i^{n+1}	new pressure at layer i	LCPNEW
R_i^n	drag at layer i	LCDRAG
A_i	area at interface i	LCAO
V_i	flow volume of layer i	LCVOLF
m_i^o	initial value of mass at layer i	LCMO
v_i^o	initial value of velocity at interface i	LCVO
d_{hi}	hydraulic diameter of layer i	-
f	two phase friction factor	-
g	gravitational constant	-
I_{vi}	implicit term in momentum equation at layer i from junction and porous flow	LCMOMI
E_{vi}	explicit term in momentum equation at layer i from junction and porous flow	LCMOME
$I_{\phi i}$	implicit term in mass and energy equations at layer i from junction and porous flow	LCIMO
$E_{\phi i}$	explicit term in mass and energy equations at layer i from junction and porous flow	LCMXO,LCEXO

3.2.6.6 Mass and Energy Equations

In Fig. 3.10b the mass and energy equations are f_m and f_e , respectively; the actual difference equations are

$$a_i^{n+1} \phi_i^{n+1} = b_i^{n+1} \phi_{i+1}^{n+1} + c_i^{n+1} \phi_{i-1}^{n+1} + d_i^n, \quad (3.32)$$

where

$$\begin{aligned} \phi_i^{n+1} &= m_i^{n+1} \text{ or } e_i^{n+1}, \\ a_i^{n+1} &= 1 + \frac{A_i \max(0, v_i^{n+1})}{V_i} \Delta t + \frac{A_{i+1} \max(0, -v_{i+1}^{n+1})}{V_{i+1}} \frac{\Delta t}{B} \\ &\quad + I_{\phi_i}^n, \\ b_i^{n+1} &= \frac{A_{i+1} \max(0, v_{i+1}^{n+1})}{V_{i+1}} \Delta t, \\ c_i^{n+1} &= \frac{A_i \max(0, -v_i^{n+1})}{V_i} \Delta t, \\ d_i^n &= \phi_i^0 + E_{\phi_i}^n \Delta t, \end{aligned}$$

and the symbols are defined in Table 3.6. Since a_i^n , b_i^n and c_i^n do not contain ϕ_i^n , the mass and energy equations are solved directly.

3.2.6.7 Pressure Equation

The fluid pressure is actually a constitutive function of the temperature and density. However, in CANDLE a more simplified approach is employed to calculate the pressure. If the molten material volume fraction is less than a specified value, ALPHL1, initially, then the pressure is equal to the region pressure, X(LCPRGN). When the initial molten material volume fraction exceeds ALPHL1, the initial value of the pressure, X(LCPO), is calculated from the Bernoulli equation with the initial velocities and upstream pressure. During the outer iteration loop the pressure is regulated to control the molten material volume fraction in regions that were initially full. That is,

$$X(LCPNEW) = X(LCPOLD) * \left(1. + \text{gain} \frac{\alpha - \alpha_o}{\alpha_o} \right), \quad (3.33)$$

where

α = current value of molten material volume fraction,
 α_o = initial value of molten material volume fraction,

and the gain is adjusted to accelerate convergence of the outer iteration.

If α is greater than α_o , then the gain is set equal to gain1 (currently gain1=0.75). If α is less than α_o and $X(LCPOLD)$ is greater than $PRATIO * X(LCPO)$, where $X(LCPO)$ is the initial value of $X(LCPOLD)$ (and currently $PRATIO=0.5$), then the gain is set equal to GAIN1, also. However, if α is less than α_o and $X(LCPOLD)$ is less than $PRATIO * X(LCPO)$, then the gain is set equal to GAIN2 (currently GAIN2=100). GAIN2 is much larger than GAIN1 because, in general, when $X(LCPOLD)$ falls below $X(LCPO)$ by a factor of $PRATIO$, the pressure is probably zero and GAIN2 accelerates the approach to zero. When $X(LCPOLD)$ falls below a specified limit, P_{low} , it is set equal to zero.

3.2.6.8 Incremental CRMESH and FLUIDS Terms

When the iteration converges, a new axial profile for the molten material mass and energy has been determined. The difference between the initial and final axial profiles defines a set of incremental changes in the mass and energy that is passed to CRMESH for use in the overall remesh algorithm. However, before this is done, CANDLE examines the final mass profile to identify regions that contain miniscule quantities that could cause numerical instabilities. If such quantities are found, the miniscule quantities are moved into neighboring regions with sufficient mass.

In addition to incremental CRMESH terms, CANDLE determines how much molten material runs off the ends of the core structure into the surrounding fluids. This material is passed to the FLUIDS module via the the LCOMPM and LCOMPT arrays. Currently, molten material is only allowed to flow into the FLUIDS domain from external type 6 candling regions, e.g., off the outside surface of the cladding at the bottom of a fuel rod. This transfer is accomplished by attaching large artificial candling

regions to the bottom of the actual candling regions in the bottom layer with non-zero interfacial areas for only the external type 6 candling regions.

3.2.7 Dynamic Mesh Treatment

3.2.7.1 General Remeshing Considerations

The dynamic mesh capabilities in the CORE module provide for an extremely flexible treatment of material relocation, phase change, mechanical deformation and chemistry. The remeshing is accomplished through four basic steps. In the first step changes that can occur on each surface of each material region are determined. In step two conflicts between various types of changes that can occur are resolved and appropriate new regions are created. In step three regions adjoining regions of the same material are combined and regions that have lost their mass are deleted. In the fourth step temperature nodes are combined or divided within a region depending on mass changes within a region.

There are six types of regions treated in CORE. Type one is a solid region that is part of the original input. It has no radial holes or cracks initially. Region type two is the same as region type one except that it has radial holes. The third type of region is a solid region created during the calculation and is typified by oxide and refrozen crust layers. Region type four is a powder and it was included specifically to treat B₄C in BWR control blades. Region type five is for material flowing between two solid surfaces (e.g., flow in the fuel-cladding gap). The final region type, region type six, is for liquid material flowing on one solid surface with a free surface.

Regions of type one, two, and three can change mass, energy and/or dimensions as a result of melting, oxidation, dissolution, breakup, hole or crack formation and/or ablation, mechanical stress, and thermal expansion. Regions of type four can change by slumping or flowing, by oxidation, by dissolution, and by melting. A region of type five can undergo oxidation, it can dissolve the surfaces it is in contact with, it can freeze and form a blockage, and its composition can change. Under certain conditions a region type five changes to a region type six. A region type six can oxidize; it can dissolve the surface it flows on; it can form a crust (a region type three) on the surface it flows on; and it can transfer liquid mass to the FLUIDS module. Table 3.6 shows the characteristics of each of the six region types.

Table 3.6						
Material Region Characteristics						
<u>Characteristic</u>	<u>1</u>	<u>2</u>	<u>3</u>	<u>4</u>	<u>5</u>	<u>6</u>
User Input	yes	yes	no	yes	no	no
Solid	yes	yes	yes	no	no/yes	no
Powder	no	no	no	yes	no	no
User Input Holes	no	yes	no	no	no	no
Calculated Holes/Cracks	yes	yes	yes	no	no	no
Created by Calculation	no	no	yes	no	yes	yes
Melting	yes	yes	yes	yes	yes	no
Dissolution	yes	yes	yes	yes	yes	yes
Erosion	yes	yes	yes	no	no	no
Oxidation	yes	yes	yes	yes	yes	yes
Mechanical Deformation	yes	yes	yes	no	no	no
Breakup	yes	yes	yes	no	no	no

3.2.7.2 Step One: Setup of Material Region Change Terms

In step one five kinds of changes are considered for each surface of a solid region. The five kinds of changes are melting, oxidation, dissolution, erosion, and breakup. For each surface, there are two surfaces for each material region, three quantities are calculated for each of the five kinds of changes that can occur. The quantities are changes in thickness, mass, and energy. In step one all calculated changes are losses from a region. Conflicts between different types of changes are resolved in step two.

Melting is the first type of change that is considered. Two situations, complete and partial melting, are treated. However, changes are only allowed if the surface temperature is equal to or greater than the solidus temperature. The melting logic is divided into two major sections to treat single and multiple heat transfer node regions respectively. (This was done because the logic requirements are much simpler for a single node than for multiple nodes.) First, the single node case will be discussed.

When there is complete bulk melting of a node and at least one surface temperature is above the solidus, the total mass of the node is available to create a region type five or six. The new region will be created on the inner surface of the current region if the current region is the first region or there isn't a solid region (region type one, two or three) as the next inner region. Otherwise the new region will be created on the outer surface of the current region. Effectively, for complete melting, the new region will be created in the same space as the old region and the old region will be given a zero thickness, mass, and energy in step two and it will be deleted in step three.

When there is partial melting, the fraction of the node mass that corresponds to the fraction of node energy that has gone through the heat of fusion is available to form a new region. This fraction is

$$\text{Frac} = \frac{H_i - H_s}{H_l - H_s} \quad (3.34)$$

where

$$\begin{aligned} H_i &= \text{enthalpy of node } i \text{ (J/Kg)}, \\ H_s &= \text{enthalpy at solidus (J/Kg), and} \\ H_l &= \text{enthalpy at liquidus (J/Kg)}. \end{aligned}$$

If both surface temperatures are above the solidus temperature, then the amount of liquid material is split equally between potential regions on each surface. However, if only one surface temperature is above the solidus temperature, then all the liquid material is available to create a new region on that surface. Figure 3.11 shows the logical flow path for single node region melting.

Multinode regions present a more difficult logic problem, although the procedure is analogous to the single node problem. As in the single node case, melting is allowed to proceed from one or both sides of the region. Also within a single time step, melting may involve more than one node and melting from both sides may involve sharing of the melt within one partially melted node in addition to fully melted nodes.

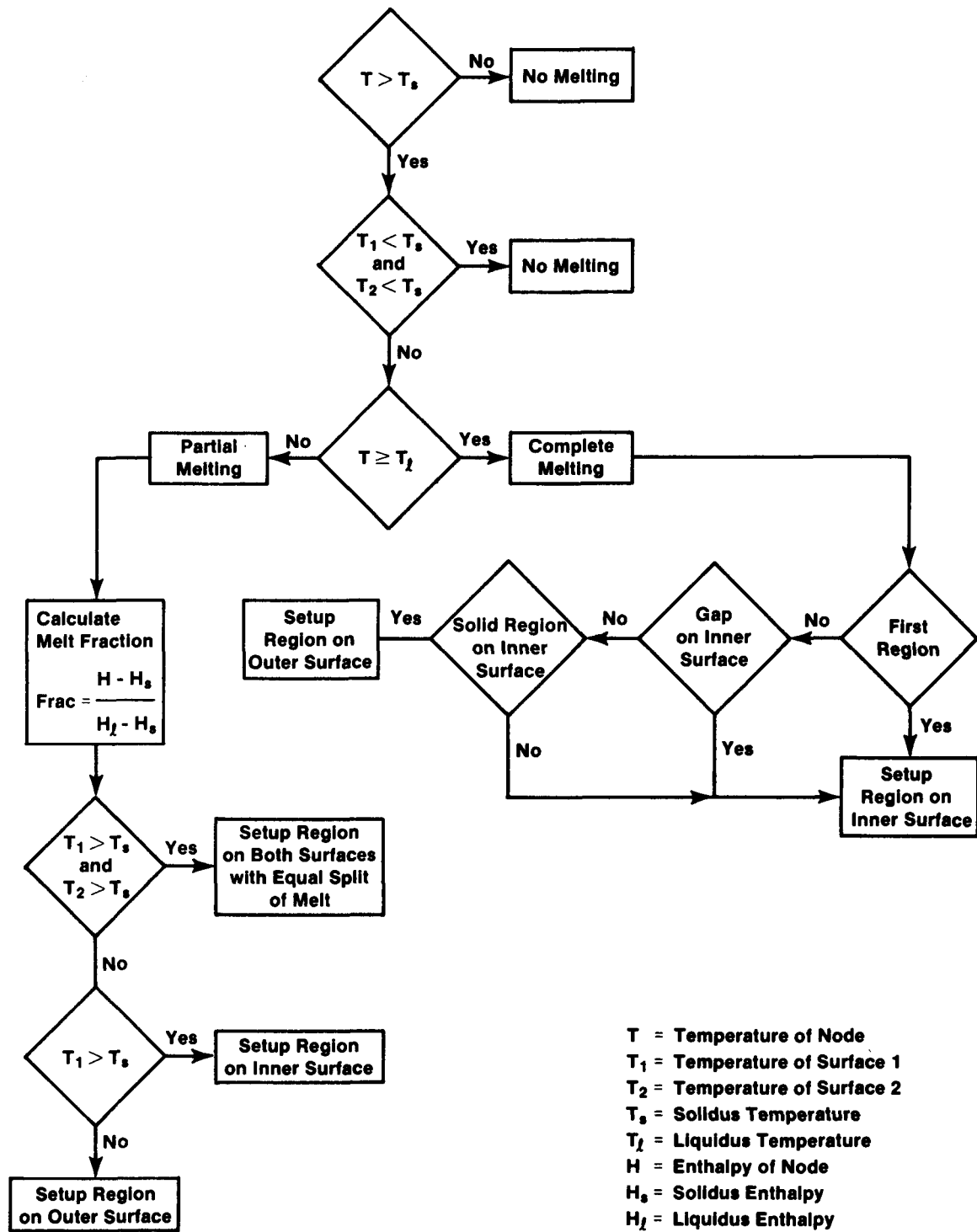


Fig. 3.11 Single node region melting logic.

After potential melting changes have been determined, changes resulting from oxidation are considered. Oxidation of zircaloy, U-Zr-O mixtures, and stainless steel are treated. (However, oxidation of stainless steel does not now result in geometry changes because material properties for the oxide have not yet been included in MELPRO.) Growth of an oxide (ZrO_2) layer on zircaloy results in thinning of the zircaloy region and the formation of a ZrO_2 region. If a ZrO_2 layer already exists the new oxide growth occurs on the interface between the zircaloy and the ZrO_2 . The new oxide growth pushes the old growth away from the zircaloy to account for the volume change on oxidation. Oxide layer growth can occur on both sides of a region.

Oxidation of U-Zr-O mixtures can occur for crusts and flowing regions, regions of type three, five, and six. If the oxidation is for a region adjoining a ZrO_2 region and the oxygen is being supplied by diffusion through the ZrO_2 region, then a new region will be created at the interface. However, if the oxygen is diffusing directly into a U-Zr-O region that can oxidize no new region will be created. Instead, the oxygen content and volume of the U-Zr-O region will be changed.

Dissolution reactions are the third form of change considered. Currently, U-Zr-O interactions with UO_2 and ZrO_2 are treated. Changes from iron/nickel interactions with zirconium and B₄C-iron interactions will be added in the near future. U-Zr-O interactions with UO_2 and ZrO_2 , or both. (In this discussion U-Zr-O can refer to zirconium, uranium, or a mixture.) Dissolution of UO_2 and ZrO_2 can occur simultaneously and, therefore, result in the potential creation of new regions on each surface, UO_2 and ZrO_2 respectively.

The final two types of changes, erosion and breakup for solid regions are not fully implemented at this time. They will be implemented as models to describe these processes for ZrO_2 and UO_2 are developed. No changes have been implemented for type four regions.

Oxidation of regions of type five and six was discussed above along with the discussion of oxidation for region types one, two, and three. However, in addition, regions of type five and six can lose mass by transfer to the FLUIDS module directly from candling. Type five regions can form blockages by freezing and type six regions can form crusts which become type three regions.

Some processes can occur simultaneously, crusting and oxidation of a type six region for example. When this happens, the masses used to create new regions must be adjusted to conserve mass and to maintain a consistent mass fraction in the old region and the region to be created. For simultaneous crusting and oxidation, the crust mass and energy are modified as follows:

$$M'_c = M_c + \frac{M_c}{M} M_o \quad (3.35)$$

$$E'_c = E_c + \frac{M_c}{M} E_o$$

where

- M'_c = adjusted crust mass,
- M_c = crust mass,
- M = region type six mass,
- M_o = mass of oxygen added during oxidation,
- E'_c = adjusted crust energy,
- E_c = crust energy, and
- E_o = energy associated with added oxygen.

In a similar manner, melting and oxidation can occur simultaneously, resulting in mass conservation problems. For example, when zircaloy cladding that is oxidizing is melting from the ZrO_2 -zircaloy interface inward and the melt reaches the inner surface an oxide region will be created on the outer surface of the zircaloy and a molten region (regions of type three and five, respectively) will be created on the inner surface. Because the zircaloy region is fully molten the same mass is involved in creating both new regions. For this case, the mass of zircaloy oxidized and its energy are used to reduce the melt mass and energy.

3.2.7.3 Step Two: Creation of Material Regions

Creation of new material regions is limited to one for each surface of each existing region, in each time step. Therefore, it would be possible for three regions to exist where one region existed at the beginning of the time step. However, because potentially more than one form of change can exist for a particular surface a hierarchical order must be established for the type of change allowed. The order chosen for step two is based on the importance of the change and the significance of delaying the change for a time step. The chosen hierarchical order is given in Table 3.7.

Table 3.7	
Hierarchical Order of the Form of Change for Each Surface	
1.	Candling
2.	Oxidation
3.	Crusting
4.	Melting, Dissolution, and Erosion
5.	Breakup

Candling has the highest priority because the candling calculational results cannot be held over from one time step to the next. Oxidation was chosen as next most important because the formation of oxide layers affects the heat transfer calculation and potential relocation of material from solid regions to those than can candle. Crusting is next because it affects the amount of material available to candle and the distance material will candle. Melting, dissolution, and erosion are all treated at the same time and from the standpoint of formation of new regions are equivalent. Breakup is treated last and it only applies to existing regions and does not create new ones.

Creation of a new region by candling, oxidation, crusting, melting, dissolution, or erosion depends on a set of criteria. For all of the above, there is a minimum mass criteria which is

$$\Delta M \geq F_3 M_z \quad (3.36)$$

where

- ΔM = mass available to form a new region (kg),
 F_3 = region formation factor user input, typically 10^{-2} - 10^{-8} ,
 and
 M_z = original node mass from region which new region is to
 be formed (kg).

This criteria only applies to situations in which there isn't an adjoining region of the type to be created. For example, if zircaloy cladding is undergoing oxidation, there must be a minimum oxide mass to create the initial oxide region; however, once the oxide region exists a new oxide region adjoining the existing one can be created with any amount of mass. Prior to formation of the initial oxide region the oxide layer thickness is

stored as part of the zircaloy region data.

In addition to the above criteria, creation of a new region by candling is only possible if it will be formed on the inner or outer surfaces or where a gap was specified in the input. Where a region of type five or six already exists, mass can flow axially without forming a new region.

Creation of an oxide region can occur provided that a candling region has not been created on the surface and the mass criteria above are satisfied. However, for a region of type five adjoining a fully oxidized crust (not a ZrO_2 region), a region of type six, or a U-Zr-O crust oxidation occurs by increasing the mass of oxygen and not by creation of an oxide region.

A crust region can only be created from a region of type six. Even though the priority for creating crusts is lower than for candling or oxidation, there is no conflict because a region type six cannot form an oxide layer, and if oxidation of the adjoining region is occurring, an oxide layer will be formed on the adjoining region.

Melting, dissolution, and erosion criteria for creating regions are all the same. These processes can form region types five or six only. A region can only be formed if the above mass criteria is satisfied and no higher priority region can be created. For these processes the mass available to form a region is determined by:

$$M_F = \text{maximum } (M_M, M_D) + M_E \quad (3.37)$$

where

M_F = mass available to form a new region,
 M_M = mass from melting,
 M_D = mass from dissolution, and
 M_E = mass from erosion.

This criteria limits the mass available to the maximum from melting and dissolution added to that from erosion.

The criteria for breakup have not been fully defined as yet. However, breakup only occurs for existing regions and does not create new regions. When breakup occurs, the mass and energy are transferred to the FLUIDS module corium fields.

Three loops are made through the region formation logic for each existing region. The first loop is used to create a region on the inner surface of an existing region. The second loop is used to modify an existing region on both the inside and outside. And the third loop is used to create a region on the outer surface of an existing region. When a new region is created, its inner and outer dimensions, mass, and energy are defined, and it is given one node.

The final part of step two involves the calculation of region and node dimensional changes for existing regions. This includes changes to the number of nodes and their masses. Six methods of recalculating dimensions are provided. These methods are given in Table 3.8. Applications of these methods to calculating dimensional changes are given in Table 3.9.

Table 3.8	
Dimensional Change Options	
<u>Method</u>	<u>Description</u>
1	Ri and Ro are known for each node and there is no swelling or strain. (Ri=inner radius and Ro=outer radius)
2	Ri is known and Ro is calculated from known mass and density.
3	Ro is known and Ri is calculated from known mass and density.
4	Ri and Ro are known without swelling and swelling is assumed to be volumetrically uniform.
5	Ri and Ro are known without strain and radii are recalculated with constant volume.
6	Ri and Ro are known without strain and radii are recalculated with constant Δr .

Finally, in step two of the remeshing component masses for each mixture region are adjusted to reflect the changes resulting from candling, melting, dissolution, erosion, and crust formation. Currently, two sets of component masses, one for fuel rods and one for control rods, are treated. The fuel rod component masses are U, Zr, O, and stainless steel while the control rod component masses are Ag-In-Cd, Zr, O, and stainless steel. For mixture regions (region types five, six, and crusts), the material

Table 3.9						
Dimensional Change Method Used for the Various Physical Situations						
<u>Physical Situation</u>	Method					
	<u>1</u>	<u>2</u>	<u>3</u>	<u>4</u>	<u>5</u>	<u>6</u>
Form Candling Region on Innermost Surface			X			
Form Candling Region in a Gap	X					
Form Candling Region on Outermost Surface		X				
Oxidation of Crusts or Type Six Regions not on Outermost Surface			X			
Oxidation of Crusts or Type Six Regions on Outermost Surface		X				
Form Oxide Region on Inner Surface of Region						X
Form Oxide Region on Outer Surface of Region						X
Oxidation Changes to Solid Region					X	
Form Crust on Outer Surface of Region Type Six	X					
Form Crust on Inner Surface of Region Type Six	X					
Change to Type Six Region	X					

Table 3.9 (cont.)						
<u>Physical Situation</u>	Method					
	<u>1</u>	<u>2</u>	<u>3</u>	<u>4</u>	<u>5</u>	<u>6</u>
Axial Flow in Region Type Five	X					
Axial Flow in Region Type Six on Innermost Surface			X			
Axial Flow in Region Type Six on Outermost Surface		X				
Form Melt Region on Inside Surface of Existing Region that is not Innermost Region	X					
Form Melt Region on Inside Surface of Existing Region that is Innermost Region			X			
Form Melt Region on Outside Surface of Outermost Existing Solid Region		X				
Change to Existing Region that is Melting				X		

properties are determined from the mass fractions of the components that are assembled in the remeshing. Future versions of MELPROG will allow for additional componet masses for each type of core structure and interactions between structures; for example, the interaction of control rod and fuel rod specific materials in PWRs and the interaction of control blade and can wall materials with fuel rods in BWRs.

3.2.7.4 Step Three: Combining and Deleting Material Regions

The creation of material regions in step two can result in two or more adjoining regions of the same type and composition. (User input regions of type one or type two are never combined.) Without a method for combining regions, the number of regions

could become very large over time. To prevent this explosion in the number of regions, a logical procedure was developed to combine similar regions and to delete regions that have lost their mass.

Regions are examined in the direction from the inside to the outside. The first region is compared to its adjoining region. If the regions are of the same type and composition, they are combined to form one region. Then the combined region is compared to the next region and the process is continued until all regions have been examined or a region that cannot be combined is found. Regions of type five and six are a special case in that when they are adjoining, they are always combined. The resulting region will be a type six region.

An additional criteria that is considered for combining regions is the existence of a gap between the regions. Gaps can exist between regions as a function of user input and can be created when a region type five is deleted. However, the number of gaps never exceeds the number in the original user input, although the makeup of the region adjoining the gap may change. If there is a gap, the gap can have zero thickness, the regions on each side of the gap will not be combined even though they are the same type and are made of the same material. An example where this criteria applies is for oxidation of the stainless steel sheath and control rod cladding in a BWR control blade (see Figure 3.5). Two oxide regions would develop and would never combine as long as they remained solid; however, if they melted, they would form a single type five region.

When a region is found that cannot be combined or all of the regions have been combined, the first or combined region is examined to determine if it should be deleted. The criteria for deletion is

$$M_r < F_3 M_z \quad (3.38)$$

where

M_r = region mass (kg),
 F_3 = user input factor, typically 10^{-2} - 10^{-8} , and
 M_z = original node mass determined when the region was initially formed (kg).

If the region is deleted, the procedure is backed up to the next inner region and then that region is checked against the next

outer region. This procedure is necessary because it is possible for a solid region separating two liquid regions to melt or dissolve resulting in two adjoining liquid regions, a situation that cannot be handled by the candling model.

3.2.7.5 Step Four: Combining and Dividing Heat Transfer Nodes

The number of nodes in a user input region (region types one, two, and four) are determined by the user. In addition, the user has control over changes to the number of nodes within input regions and some created regions. Three input parameters are involved in deciding to subdivide a node, combine nodes, and limit a region to a single node. (In this discussion, node refers to material nodes only and does not include surface nodes. See Section 3.2.2 for a detailed definition of material and surface nodes.) A node is combined with its neighbor if the nodes mass meets the following criteria

$$M_i < F_1 \cdot M_z \quad (3.39)$$

where

M_i = mass of node i (kg),
 F_1 = combination factor, typical value of 0.1, and
 M_z = original node mass determined when region was initially formed (kg).

This means that using the typical value of 0.1 for F_1 , a node that has a mass less than 10% of its original mass, would be combined with a neighboring node. (The region would have to have more than one node at the beginning of step four.)

One node is divided into two or more equal mass nodes if the following criteria is satisfied.

$$M_i > F_2 \cdot M_z \quad (3.40)$$

where

M_i = mass of node i (kg),
 F_2 = node dividing factor, typical value is 2.0, and
 M_z = original node mass determined when region was initially created (kg).

The third criteria is used to limit regions to a single node. Regions of type five and six are always limited to one node.

Type three regions are limited to a single node if they are formed by crusting from a region of type six. However, a region of type three that results from formation of an oxide layer can have more than one node, if the oxide layer has sufficient mass. The user sets a flag for each region of a core structure that is input. If this flag (ICNODS) is set to zero for a region, the number of nodes in the region will be limited to one. If other input specifies multiple nodes in the region, the multiple nodes will exist until the first time the core structure is remeshed. Then, all nodes in the region will be combined. Regions created from an existing region are given the value of ICNODS of the existing region; however, the above limitations relating to region types are always in effect. The primary purpose of this flag is to allow the user to prevent subdividing of nodes.

Whenever nodes are combined or subdivided, the mass and energy are conserved; however, temperature information is lost and temperature gradients are not conserved. Temperatures on region surfaces are conserved, however.

Step four also sets power factors for created regions, creates cracks in regions, and sets up special regions for axial sections of core structures that have lost all of their mass. At this time, power factors are only set for fuel rods and then only for UO_2 relocation. The power factor is set for region types three, five, and six, corresponding to a crust made up of U-Zr-O, an internal candling region, and external candling region, respectively. The power factor is calculated as

$$Pr = \frac{M_u(238.+32.)}{238. M_u + M_{Zr} + M_o + M_{ss}} \quad (3.41)$$

where

Pr = region power factor,
 M_u = mass of uranium (kg),
 M_{Zr} = mass of zirconium (kg),
 M_o = mass of oxygen (kg), and
 M_{ss} = mass of stainless steel (kg).

This version of the CORE molecule includes a simple parametric model for failure of oxide layers and crusts on fuel rods. The model assumes that a 20.0° split over the axial length of an axial layer will occur if both a temperature criteria and thickness criteria are satisfied. The temperature criteria is

input by the user and is satisfied when the inner surface temperature of the region exceeds the input failure temperature.

Axial layers can completely disappear by melting, dissolution, and breakup. When this happens, two sections of a core structure can become separated. At this time, models for relocating non-liquid core structure components have not been developed except for transfer of material to the solid corium field in the FLUIDS module. A special type six region exists to provide a coupling between fuel rod stubs hanging from the upper core grid plate and fuel-rod stubs projecting upward from the lower core grid plate. This region can have zero mass and it provides a path for candling between the lower and upper ends of the rod stubs, respectively. Material in transit through this region is treated by the candling model only. All other physical models are turned off for this region and axial layer.

3.3 References

- 3.1 R.D. Richtmyer and K.W. Morton, Difference Methods for Initial Value Problems, Second Edition, Interscience Publishers, New York, NY (1967).
- 3.2 J.T. Prater and E.L. Courtright, "High Temperature Oxidation of Zircaloy-4 in Steam and Hydrogen Environments," NUREG/CR-4476, PNL-5558, Pacific Northwest Laboratory, Richland, WA (1986).
- 3.3 J.F. White, et al., "Seventh Annual Report - AEC Fuels and Materials Development Program," GEMP-1004 (1968).

4.0 STRUCTURES MODULE

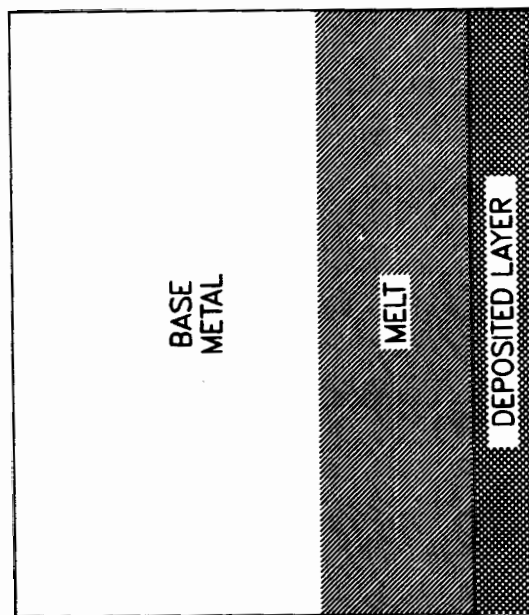
The STRUCTURES module consists of two major parts: heat transfer and mechanical models. A description of the heat transfer models is given in Sections 4.1 through 4.7 and the mechanical models are discussed in Sections 4.8 through 4.10. A brief description of the major subroutines in the STRUCTURES module is given in Appendix A.

4.1 Structure Heat Transfer

The structure heat transfer module is intended to calculate all heat transfer processes for the reactor vessel and all in-vessel structures excluding the reactor core (i.e., fuel and control rods). The intent is to treat both PWR and BWR designs; however, MELPROG-PWR/MOD1 is limited to PWR designs. The module has been designed to treat both one-dimensional and two-dimensional conduction heat transfer, bulk melting, crust formation and bulk freezing, and plateout and deposition of fission products and aerosols. In the PWR/MOD1 version, one-dimensional thermal conduction and bulk melting are treated. Crust formation and freezing of bulk fluid components onto structures are currently under development and will be incorporated in later releases of MELPROG. Two-dimensional conduction heat transfer will also be provided for certain types of structures in the future. Boundary conditions to the two-dimensional FLUIDS (steam and/or hydrogen, water, and corium) module, to the radiation heat transfer module, and to the DEBRIS module are also treated in the current version.

4.2 Structure Types

The in-vessel structures are categorized as walls, plates, columns, and general structures as listed in Table 4.1. Walls in a PWR are the core baffle, the core barrel, the thermal shield, and the vessel wall. PWR plates include the vessel top and bottom heads, the lower core support plate, the lower core grid plate, the upper core grid plate, the upper support plate, and others. Special structures currently include columns and a general structure. Each of these categories can be expanded to treat additional specific structures, for example BWR structures. Typical plate and wall type structures are shown in Fig. 4.1.



1D HEAT TRANSFER

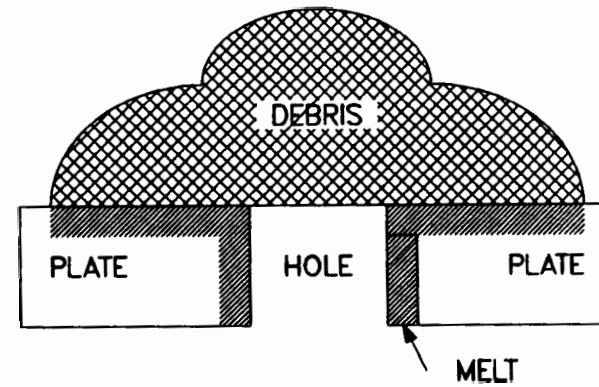


PLATE ABLATION

Figure 4.1 Typical wall and plate type structures

Table 4.1			
Examples of Structure Types			
<u>Wall</u>	<u>Plate</u>	<u>Column</u>	<u>General</u>
Core Baffle	Vessel Bottom	Support Colum	Upper Plenum Structure
Core Barrel	Lower Core Support Plate	Guide Tubes	Steam Separators
Thermal Shield	Diffuser Plate		
Vessel Wall	Lower Grid Plate		
	Upper Grid Plate		
	Upper Core Support Plate		
	Vessel Head		

Structures are set-up on a two-dimensional cell basis. Currently a maximum of five structures can be input in a particular cell and a maximum of six structures can be treated. The extra space is provided to treat movement of failed structures from one cell to another. The limits (five input and six total) can be easily changed and will be made part of the input in future versions of MELPROG.

4.2.1 Wall Type Structures

All current wall structures are annuli. These may be placed at either the inner, the outer, or at both the inner and outer radial surface(s) of the computational cell as illustrated in Fig. 4.2. Surface 1 of a wall always faces toward the cell center and always uses the fluid conditions (i.e., the temperature and heat transfer coefficient) in the cell to formulate its boundary conditions. Surface 2 always coincides with the cell boundary and always faces the adjoining cell or the problem boundary. If surface 2 faces the adjoining cell, then the fluid conditions in the adjoining cell are used to formulate its boundary condition. If surface 2 is at the problem boundary, three optional boundary conditions are available. First, it may use the fluid conditions in its cell to formulate its boundary condition. Second, it may use an adiabatic boundary condition.

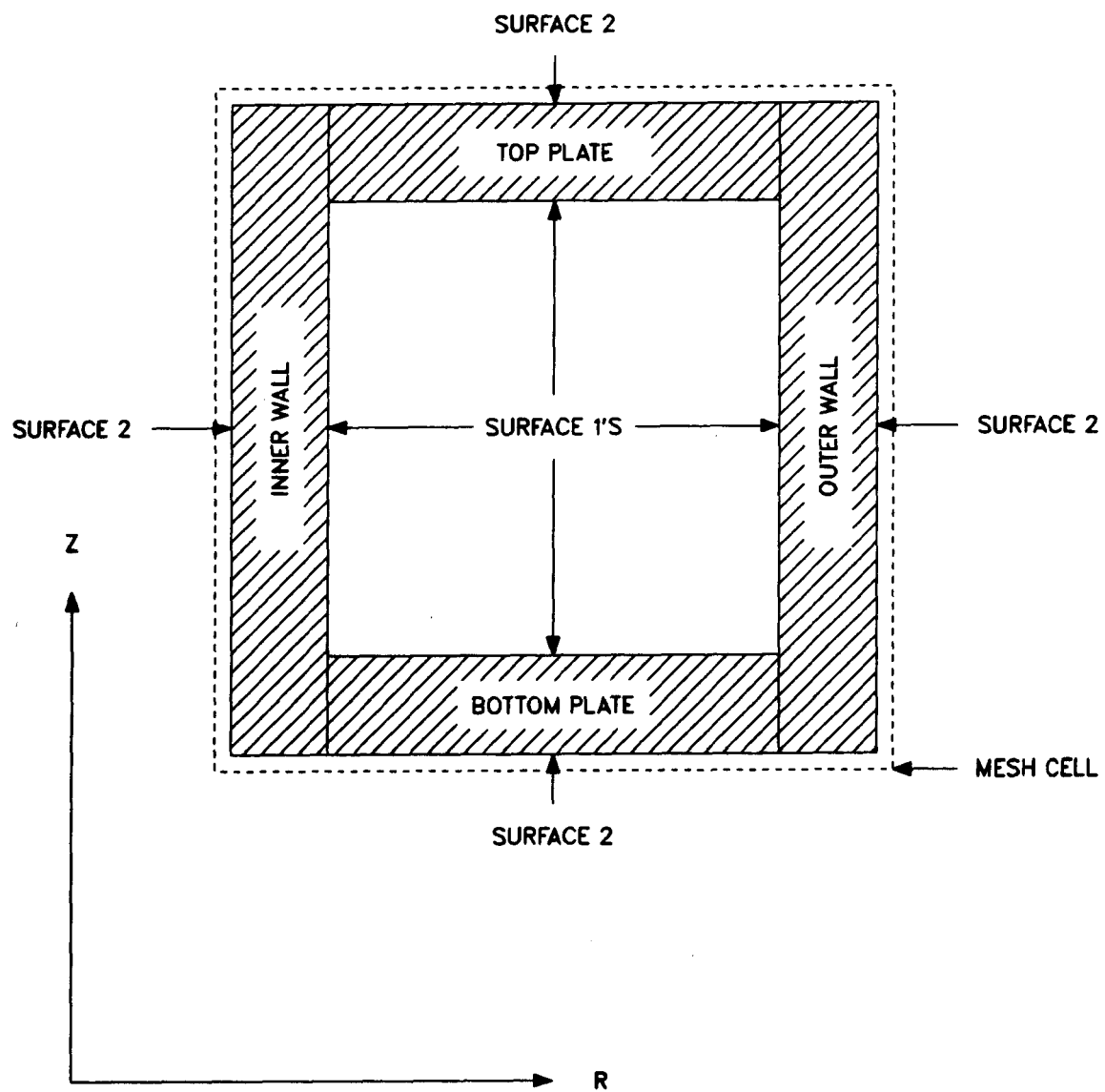


Figure 4.2 Structure arrangement in a FLUIDS computational cell

Third, it may use an input, constant sink temperature with an input, constant heat transfer coefficient. If the radiation heat transfer module is activated, then radiation heat transfer is calculated for both surfaces unless surface 2 is adiabatic. Currently walls do not have holes in them.

4.2.2 Plate Type Structures

All plates are assumed to be horizontal and flat for heat transfer calculations. Plates may or may not have holes depending on the input. In PWR/MOD1 all holes in plates are assumed to be round. In the future, rectangular holes will be treated explicitly. A plate can be input at the bottom, the top, or at both surface(s) of a computational cell. Surface 1 of a plate always faces into the cell and always uses the fluid conditions in the cell to formulate its boundary conditions. Surface 2 always faces the neighboring cell or the problem boundary and is located on the cell boundary. As in the case of walls, if surface 2 faces an adjoining cell, it uses the fluid conditions in the adjoining cell to formulate its boundary condition. Likewise, if surface 2 is at the problem boundary its boundary condition may either use its cell's fluid conditions, be adiabatic, or use an input, constant sink temperature with an input, constant heat transfer coefficient. The one-dimensional heat transfer model for plates with holes assumes that half of the hole surface area is part of the upper surface and the other half is part of the lower surface. The thickness of the plate for heat transfer is the plate's actual volume divided by half of its total surface area. If activated, radiation heat transfer is treated on both surfaces except when surface 2 is specified to be adiabatic in which case only surface 1 radiates. If a debris bed rests on a plate, then surface 1 uses the debris conditions to formulate its boundary conditions.

4.2.3 Column Type Structures

A column is a vertical cylindrical structure that may or may not be hollow; however, it must always have a finite inside radius. Its outer surface is surface 1 and it sees the fluid conditions in its computational cell. The inner surface is surface 2 and it may be adiabatic (for example a solid cylinder) or it may see the same fluid conditions as surface 1. In the current version, there are no radiation heat transfer boundary conditions on columns.

4.2.4 General Type Structures

A general type structure is a slab with no particular geometric orientation or location in a computational cell. The structure is defined by a heat transfer thickness and a total surface area. The surface area is divided into equal area surfaces 1 and 2. Surface 1 always sees the cell's fluid conditions and surface 2 may either see the cell's fluid conditions, be adiabatic, or see an input, fixed boundary temperature through an input, constant heat transfer coefficient. General structures do not have radiation heat transfer boundary conditions. General structures are useful for treating the heat capacitance of structures that have complicated geometries which are not explicitly modeled in MELPROG. They are also useful for treating structures for which only the mass and surface area are known. General structures should not be partitioned across fluid cell boundaries.

4.3 Mesh Arrangement

Any structure may be made up of one or more materials. Each material is treated by at least one region; however, a single material may be modeled by more than one region. A region is defined by its material properties, its dimensions, and by the number of heat transfer nodes in it. Within a region, heat transfer nodes are sized to form equal masses. Temperatures are calculated at region surfaces and at the mass center of each node.

When complete melting of a structure in a cell occurs, the mass in that cell is transferred to the corium fields of the FLUIDS module. The structure is then removed from the problem and all coupling to that structure is deleted. This means that the radiation heat transfer view factors in that cell are recalculated and that axial and radial flow areas and hydraulic diameters are recalculated, for example. In the future some of the dynamic mesh capabilities discussed in Section 3.0 for core structures will be added to the treatment of ex-core structures.

4.4 Material Properties

Temperature-dependent thermal conductivity and specific heats are calculated in the MELPRO subroutine that is discussed in Section 10.

4.5 Conduction Heat Transfer Solution Technique

4.5.1 Formulation for Material Nodes

A modified Crank-Nicolson finite difference scheme [4.1] is used for both the one and two-dimensional heat transfer formulations for material nodes. This formulation is:

$$\begin{aligned} \frac{(mC_p)_i^n (T_i^{n+1} - T_i^n)}{\Delta t} = & (1 - \beta) \sum_{j=1}^N \left[k_{ij}^n \left(\frac{A}{\Delta x} \right)_{ij}^n (T_j^{n+1} - T_i^{n+1}) \right] \\ & + \beta \sum_{j=1}^N \left[k_{ij}^n \left(\frac{A}{\Delta x} \right)_{ij}^n (T_j^n - T_i^n) \right] \\ & + m_i^n Q_i^n \end{aligned} \quad (4.1)$$

where

T_i^{n+1} = temperature in node i at time step n + 1 (K),

T_i^n = temperature in node i at time step n (K),

C_{pi}^n = the specific heat of node i at time step n (J/kg),

m_i^n = the mass of node i at time step n (kg),

Δt = the time step (s),

β = the forward weighting factor
 $0 \leq \beta \leq 0.49999$,

N = the total number of temperature nodes,

k_{ij}^n = the thermal conductivity between nodes i and j at time step n (W/m K),

$\frac{A}{\Delta x_{ij}^n}$ = the heat transfer geometry factor between nodes i and j at time step n (m), and

Q_i^n = the power per unit mass in node i at time step n (W/kg) .

The forward weighting factor (β), obtained from a linear stability analysis, is calculated as follows:

$$\beta = \text{minimum } (\beta_1, \beta_2, \dots, \beta_i, \dots, \beta_N)$$

where
$$\beta_i = \frac{(mCp)_i^n / \Delta t}{\sum_{j=1}^N k_{ij}^n \left(\frac{A}{\Delta x} \right)_{ij}^n}$$

4.5.2 Surface Nodes

Surface nodes or temperatures are treated somewhat differently because they have no heat capacitance. The formulation for these nodes is:

$$0 = \sum_{j=1}^N \left[k_{ij}^n \left(\frac{A}{\Delta x} \right)_{ij}^n (T_j^{n+1} - T_i^{n+1}) \right] + A_i Q_i^n \quad (4.2)$$

where the terms are the same except for

Q_i^n = the surface heat flux, usually radiation heat transfer (W/m^2), and

A_i = the surface area of node i (m^2).

At the boundaries, only the surface node equation is involved. Radiation heat transfer is treated as a surface source. When heat transfer to the fluid components (steam and/or hydrogen, water, and corium in MOD1) is involved the surface node equation becomes:

$$\begin{aligned} 0 = & H_{gs}^{n+1} A_i^n (T_g^{n+1} - T_i^{n+1}) + H_{ls}^{n+1} A_i^n (T_l^{n+1} - T_i^{n+1}) \\ & + H_{css}^{n+1} A_i^n (T_{cs}^{n+1} - T_i^{n+1}) + H_{cls}^{n+1} A_i^n (T_{cl}^{n+1} - T_i^{n+1}) \\ & + \sum_{j=1}^N \left[k_{ij}^n \left(\frac{A}{\Delta x} \right)_{ij}^n (T_j^{n+1} - T_i^{n+1}) \right] + A_i Q_i^n \end{aligned} \quad (4.3)$$

where the terms are as defined above except as follows

H_{gs}^{n+1} = the heat transfer coefficient from steam and/or hydrogen to structure at time step $n+1$ ($W/m^2 K$),

T_g^{n+1} = the temperature of gas at time step $n+1$ (K),

H_{ls}^{n+1} = the heat transfer coefficient from water to structure at time step $n+1$ ($W/m^2 K$),

T_l^{n+1} = the temperature of water at time step $n+1$ (K),

H_{css}^{n+1} = the heat transfer coefficient from solid corium to structure at time step $n+1$ ($W/m^2 K$),

T_{cs}^{n+1} = the temperature of solid corium at time step $n+1$ (K),
 H_{cls}^{n+1} = the heat transfer coefficient from liquid corium to structure at time step $n+1$ ($w/m^2 K$), and
 T_{cl}^{n+1} = the temperature of liquid corium at time step $n+1$ (K).

Heat transfer between corium and structures assumes a dispersed particle flow regime for solid corium. Currently, no heat transfer coefficient is calculated for liquid corium. Heat transfer from corium to structures also is treated through thermal radiation and through conduction from debris beds. The transfer coefficients described above are supplied by the FLUIDS module. For debris bed to structure heat transfer, the heat transfer coefficient is supplied by the DEBRIS module.

4.5.3 Matrix Solution

For one-dimensional heat transfer the equations can be arranged to form a tridiagonal matrix that is easily solved. A simple one material node example of the equations in tridiagonal form is:

$$\begin{aligned}
 C_{11}T_1^{n+1} + C_{12}T_2^{n+1} &= R_1 \\
 C_{21}T_1^{n+1} + C_{22}T_2^{n+1} + C_{23}T_3^{n+1} &= R_2 \\
 C_{32}T_2^{n+1} + C_{33}T_3^{n+1} &= R_3
 \end{aligned} \tag{4.4}$$

where $C_{11} = k_{12} \left(\frac{A}{\Delta x} \right)_{12} + A_1 (H_{gs} + H_{ls} + H_{css} + H_{cls})$

$$C_{12} = -k_{12} \left(\frac{A}{\Delta x} \right)_{12}$$

$$R_1 = A_1 \left(H_{gs} T_g^{n+1} + H_{ls} T_l^{n+1} + H_{css} T_{cs}^{n+1} + H_{cls} T_{cl}^{n+1} \right) + A_1 Q_1$$

$$C_{21} = (1-\beta)C_{12}$$

$$C_{22} = \frac{(mC_p)_2}{\Delta t} - C_{21} - C_{23}$$

$$C_{23} = -(1-\beta) k_{23} \left(\frac{A}{\Delta x} \right)_{23}$$

$$R_2 = \beta \left[-C_{12}(T_1^n - T_2^n) - C_{23}(T_3^n - T_2^n) \right] \\ + m_2 Q_2 + \frac{(mC_p)_2}{\Delta t} T_2^n$$

$$C_{32} = C_{23} / (1 - \beta)$$

$$C_{33} = -C_{32} + A_3(H_{gs} + H_{ls} + H_{css} + H_{cls})$$

and

$$R_3 = A_3(H_{gs}T_g^{n+1} + H_{ls}T_l^{n+1} + H_{css}T_{cs}^{n+1} + H_{cls}T_{cl}^{n+1}) + A_3 Q_3$$

Extension of this equation to many materials and multinode cases is straightforward.

4.6 Phase Transition Heat Transfer

Phase transitions include melting, freezing, and solid-solid transitions. The solution procedure involves an iteration on the finite difference temperature solution. All phase transitions are assumed to occur over a temperature range of 1.0 K or more. During the iteration an "effective" specific heat is used in the temperature calculation. The "effective" specific heat is a combination of the phase transition energy and the actual specific heat.

The phase transition iteration loop consists of three sections. The first part is for phase transitions with decreasing temperatures. The second part is for phase transitions with increasing temperatures. And the third section is for both types of transitions. In parts one and two the net energy change in each node undergoing a phase transition and a corrected temperature are calculated. For increasing temperature conditions the net energy change is

$$Q_i = C_{P_{eff}_i} (T_i^{n+1} - T_i^n) \quad (4.5)$$

where

Q_i = net energy change in node i (J),

$C_{P_{eff}_i}$ = "effective" specific heat of node i (J/K)

$T_i^{n+1,k}$ = temperature of node i at time step n+1 and iteration k (K), and

T_i^n = temperature of node i at time step n (K).

The adjusted temperature is calculated as follows. First, if T_i^n is less than the threshold temperature for the phase transition the value of Q_i is adjusted for the energy needed to raise the temperature to the threshold. That is, Q_i' is given by:

$$Q_i' = Q_i + C_{p_i}(T_i^n - T_{1i}) \quad (4.6)$$

where

C_{p_i} = true specific heat of node i (J/K) and

T_{1i} = threshold temperature for the phase transition (K).

The adjusted temperature is then calculated as

$$T_i^{n+1,k'} = Q_i'/SC + \text{maximum}(T_i^n, T_{1i}) \quad (4.7)$$

where

$SC = C_{p_i} + H_{\text{phase}_i}$ and

H_{phase_i} = phase change energy expressed in units of J/K.

If $T_i^{n+1,k'}$ is greater than the upper phase transition temperature a further adjustment is necessary. The adjusted temperature is then calculated as follows:

$$Q_i' = Q_i' + SC \left[\text{maximum}(T_i^n, T_{1i}) - T_{hi} \right] \quad (4.8)$$

where

T_{hi} = upper phase transition temperature for node i.

and

$$T_i^{n+1,k'} = Q_i' / C_{p_i} + T_{hi}.$$

In parts one and two there is also a second procedure used to calculate the energy change in each node undergoing a phase transition. This procedure assumes that the temperature $T_i^{n+1,k}$ is correct and that the energy change in the node must be adjusted.

The adjusted energy for conditions in which the node temperature is increasing is calculated as:

$$Q_p = C_{p_i} (T_i^{n+1,k} - T_i^n) + H_{\text{phase}_i} \left[\text{maximum} \left[T_{l_i}, \text{minimum} \left(T_i^{n+1,k}, T_{h_i} \right) \right] - \text{minimum} \left[T_{h_i}, \text{maximum} \left(T_i^n, T_{l_i} \right) \right] \right] \quad (4.9)$$

Completely analogous procedures are used to calculate $T_i^{n+1,k'}$ and Q_{p_i} for conditions in which the node temperature is decreasing. In part three these results are used to calculate two new "effective" specific heats as follows:

$$C_{p_{\text{eff}_{i,1}}} = Q_{p_i} / (T_i^{n+1,k} - T_i^n) \quad (4.10a)$$

and

$$C_{p_{\text{eff}_{i,2}}} = Q_i / (T_i^{n+1,k'} - T_i^n) \quad (4.10b)$$

If $T_i^{n+1,k'}$ is less than T_{l_i} or $T_i^{n+1,k}$ is greater than T_{h_i} , then $C_{p_{\text{eff}_{i,1}}}$ is used in the next phase transition iteration otherwise $C_{p_{\text{eff}_{i,2}}}$ is used. If convergence is not achieved in MRELAX (currently fifteen) iterations, a relaxation procedure is used to continue the iteration.

When triggered, the relaxation procedure constrains $C_{p_{\text{eff}_i}}^{k+1}$ to

$$C_{p_{\text{eff}_i}}^{k+1} = \text{minimum} \left[\text{CHANGP} * C_{p_{\text{eff}_i}}^k, \text{maximum} \left(\text{CHANGM} * C_{p_{\text{eff}_i}}^k, \frac{C_{p_{\text{eff}_{i,1}}} + C_{p_{\text{eff}_{i,2}}}}{2.0} \right) \right] \quad (4.11)$$

where

CHANGP = 1.0 + RELAXO initially, and

CHANGM = 1.0 - RELAXO initially.

Cp_{effi}^{k+1} is also constrained to the range between $Cp_{effi,1}$ and $Cp_{effi,2}$. Very NRELAX iterations (currently 3), the values of CHANGP and CHANGM are recalculated as

$$CHANGP = (CHANGP-1.0)*RELAXN+1.0 \text{ and}$$

$$CHANGM = 1.0 - (1.0 - CHANGM)*RELAXN.$$

The values of CHANGP and CHANGM are adjusted until $CHANGP-1.0$ is less than $STCNVG*RELAXM$. STCNVG is the phase change convergence criteria (user input) and RELAXM is a factor (currently set to 5.0).

4.7 Structure Oxidation

Stainless steel and zircaloy structures can oxidize on their outer surfaces. The parabolic rate equations used to calculate structure oxidation are the same as those discussed in Section 3.2.3.1 for the CORE module. However, unlike for core structures, material regions made of oxidized material are not created and the impact of formation of an oxide layer is not considered on the conduction of heat into or out of a structure. The energy produced by oxidation is deposited on the structure surface. Structures that reside on fluid cell boundaries (walls and plates) use steam from the adjoining fluid cell for side 2 and steam from the fluid cell that the structure resides in for side 1. Similarly, the hydrogen source term provided to the FLUIDS module accounts for the location of structures on fluid cell boundaries. Structures other than walls and plates take steam from and supply hydrogen to the fluid cells in which they reside.

4.8 Structural Mechanics Modeling

The objective of the structural mechanics models is to predict failure of major in-vessel structures due to mechanical loading. The specific structures to be considered fall generally into three categories: cylindrical walls, flat perforated plates supported at regular intervals, or elliptic plates supported at the edges, with or without perforations. There are also several special structures that need to be considered, such as support columns (Westinghouse PWR) or support girders (Babcox & Wilcox PWR), headbolts, etc., whose presence depends on the particular reactor design. The current structures mechanics module contains

simplified models describing the three general categories mentioned above, plus a support column model, which is specific to the Westinghouse PWR design.

4.9 Modeling Design Consideration

There are several factors present in reactor designs that guided the selection of the mechanical failure models. The most important factor is that most in-vessel structures, while possibly supporting great weight, are actually under very little stress. Most structures are under little stress because reactors are designed to minimize deformation of internal support structures under normal operating conditions. The most prominent exception to the low stress assumption is the vessel itself, which is under high internal pressure. Associated components under high stress are the head bolts. The low stress implies that structures must be at high temperatures for failure to occur under most conditions, suggesting that a plastic or creep failure criterion is applicable. Failure at low temperature could occur if a structure were subjected to a transient high pressure, as for instance from a steam spike or vapor explosion. Such a pressure transient could cause a ductile-fracture type of failure, with the most probable location for such a failure being either in the lower plenum or at the headbolts [4.2].

Another factor in the choice of mechanical models was the desire to avoid detailed calculation of stress-strain distributions in the structures. Calculation of inelastic mechanical behavior can be time-consuming in terms of computer resources, usually requiring a finite element or finite difference solution of the mechanics equations.

The approach taken for MELPROG/MOD1 is to use stresses calculated at known peak stress locations by analytic formulae. Since very little structural deformation takes place under the given low loading conditions, the stresses may be calculated by elastic equations for most of an accident sequence. Even when yield conditions are reached, the structure remains close to its initial geometry until failure is imminent. The peak stress locations can then be analyzed using the simple stress distributions present under fully plastic conditions, rather than calculating a detailed elastic-plastic distribution.

The mechanics models chosen predict only the stress and not the strain; this requires choosing a failure criterion that does

not depend on strain, but that considers both plastic behavior and high-temperature creep. A well-known criterion satisfying this requirement is the Larson-Miller Parameter (LMP) correlation [4.3], which relates time-to-failure to the stress and temperature of a structure. The LMP correlation also has the advantage of allowing easy extrapolation of rupture times at a given stress and temperature to other temperatures, which was felt to be desirable due to the limited database available. The LMP correlation was chosen as the main failure criterion in MELPROG/MOD1.

The life fraction rule is used in conjunction with the LMP correlation to account for the changing load and temperature conditions with time. The LMP correlation in SI units and associated life fraction rule are

$$LMP = T[\log_{10}(t_R) - 16.44], \quad (4.12a)$$

$$\int_0^{t^*} \frac{dt}{t_R} = 1, \quad (4.12b)$$

where

LMP = the Larson-Miller Parameter,
 T = the temperature (K),
 t* = the total time to failure (sec), and
 t_R = the time to rupture (sec).

4.10 Mechanics Modeling

The following section describes the models used for the different types of structures in the Westinghouse PWR, including some examples of typical stresses. First, it is helpful to show a realistic example of the structures to be modeled and their relationship to each other. A stylized representation of the major Westinghouse in-vessel structures is shown in Fig. 4.3. The core sits on a lower core grid plate, which in turn sits on support columns. These columns are supported by a lower core support plate, which hangs by its edge from the core barrel. The core barrel is attached at its upper edge to the head flange. The vessel bottom is supported at its edge by the vessel wall. The vessel wall itself is supported at the head flange by external support structures, which do not concern us here. The top core grid plate hangs on support columns from an upper core support plate, which is supported by its rim from the head flange.

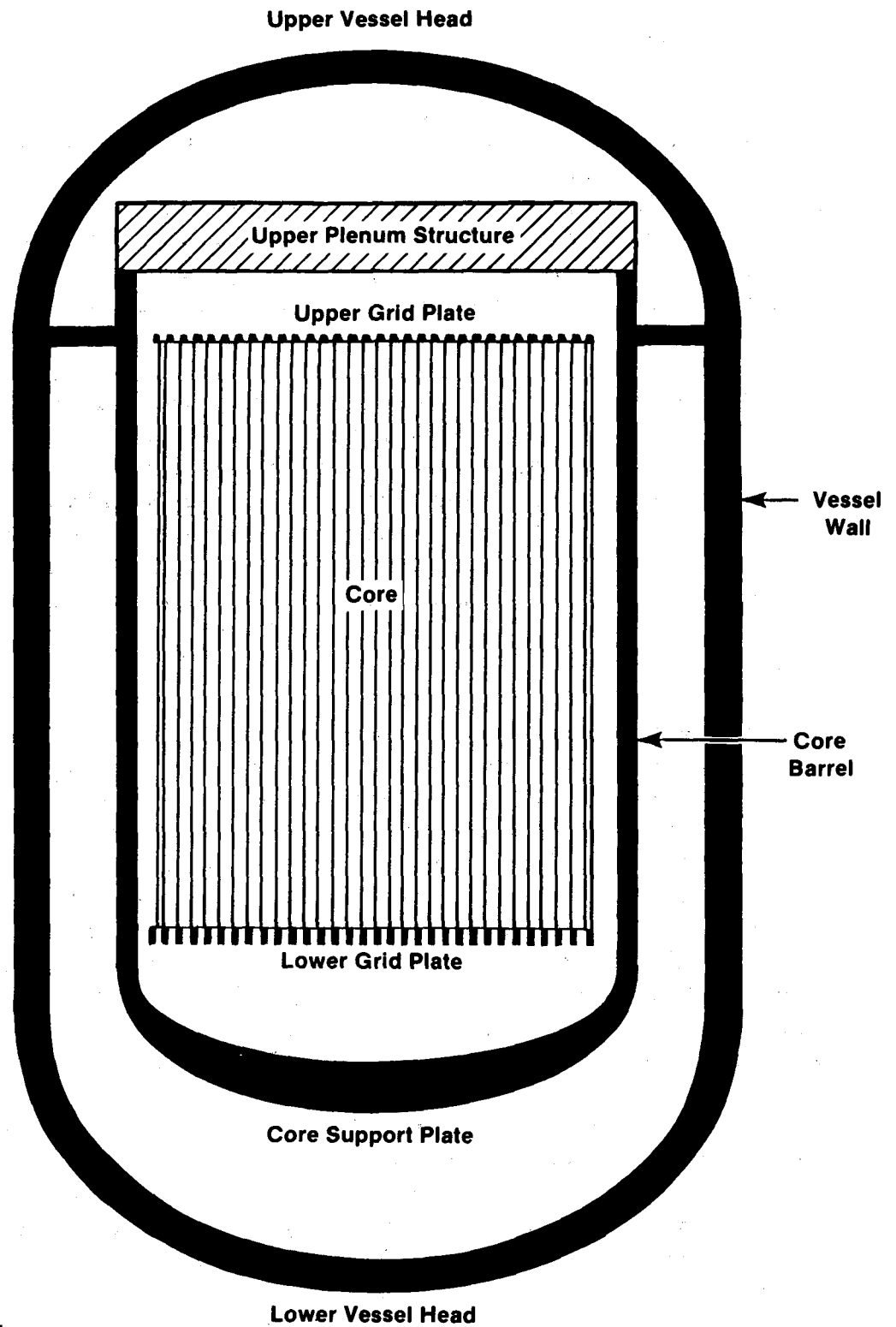


Figure 4.3 Westinghouse PWR schematic

4.10.1 Lower Core Grid Plate

The grid plate is supported at regular intervals by support columns in a square pattern. It is perforated by holes, also in a square pattern (Fig. 4.4). The presence of the holes can be accounted for by modifying the mechanical properties of the plate and then using analysis methods for a solid plate with these "equivalent" mechanical properties [4.4,4.8]. In the context of plastic failure, the major effect of the holes is to reduce the load-bearing cross-sections.

The plate can be treated by analyzing one cell, consisting of four columns and the associated square section of the plate [4.5]. The procedure used is to find the moments at the center and corners of this cell, using elastic solutions for a flat plate supported at the corners by columns with the mechanical properties adjusted for the presence of the holes. These elastic moments are used to calculate the stresses at the plate surfaces, assuming a linear (elastic) distribution of stress through the plate thickness.

The elastic equations are strictly applicable only until the moment reaches the moment of incipient yield M_y , corresponding to the point at which the stress at the plate surface reaches the yield stress σ_y . When incipient yield is reached, the plate does not deform significantly, but the stress distribution through the plate thickness now consists of a linear (elastic) segment and two segments next to the plate surfaces which are at the yield stress (plastic). In MELPROG, the elastic equations are used during this elastic-plastic range with a correction factor derived from the plastic theory of beams [4.9] to account approximately for the change in the stress distribution.

The stress used for the LMP criterion is taken to be either twice the average shear stress or equal to the principal stress, depending on which is larger in magnitude. The average shear stress is equal to one-half the peak (surface) stress while the peak moment from the elastic equations is less than M_y . In MELPROG, a correction is made by dividing the moment (and the stress) by 1.5 (the ratio of M_p to M_y) and using one-half of the resulting stress as the average shear. This procedure is done to account approximately for the nonlinear stress distribution in the plate at moments between M_y and M_p , and, although the stress in the plate is reduced below the actual stress value for the

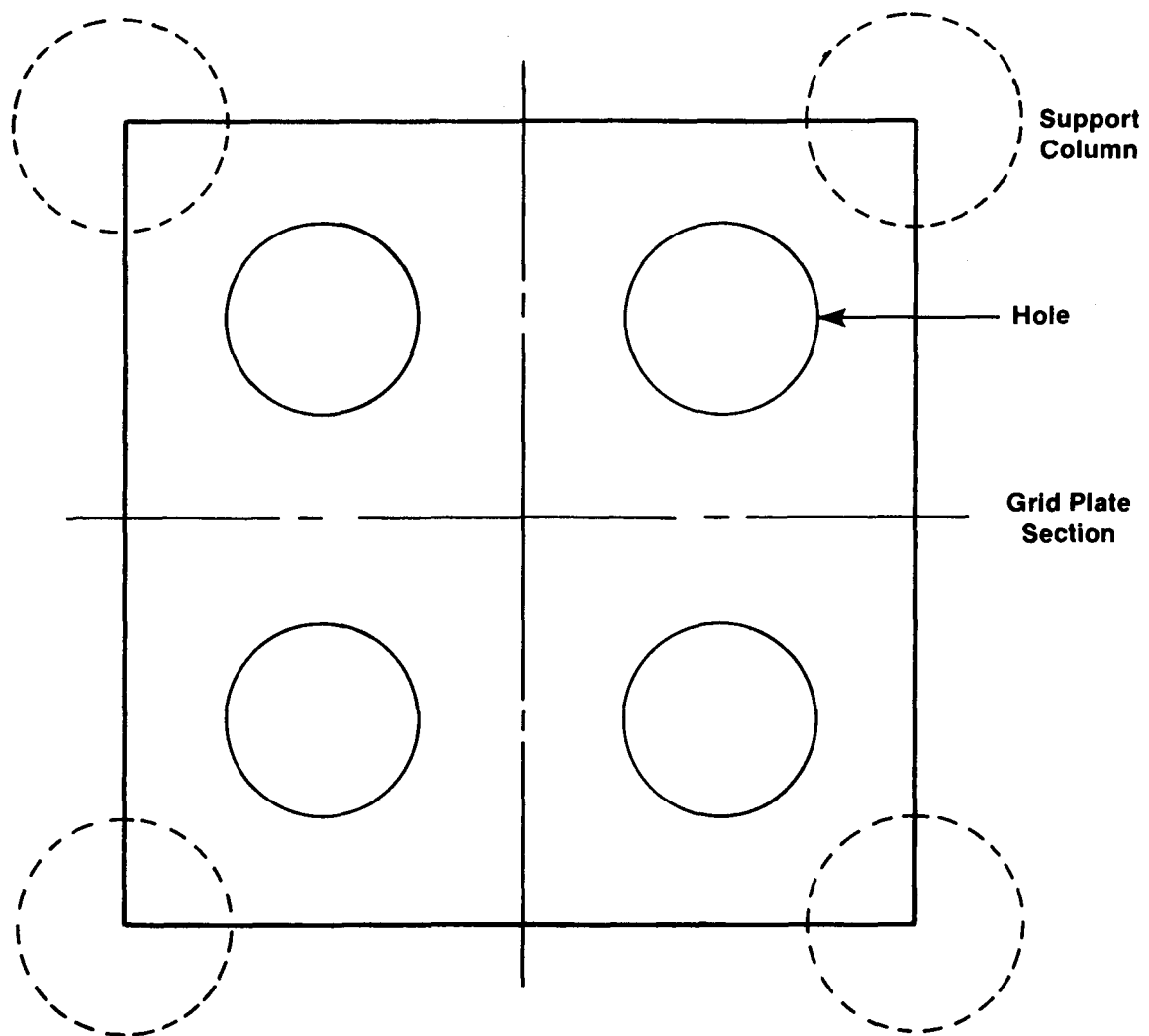


Figure 4.4 Cell of Westinghouse lower core
grid plate, top view

pure elastic case, the elastic part of the loading sequence is not important to plate failure, and the reduction is of no consequence.

The net effect of the correction factor is to reduce the stress applied to the LMP criterion to only $0.667\sigma_Y$, rather than σ_Y , when the moment reaches M_Y . When M_p is reached (imminent failure), the applied stress is equal to σ_Y . Assuming that the LMP criterion is consistent with plastic failure, failure should be predicted under these stress conditions.

The average plate temperature is used in the LMP correlation, so that temperature distributions through the plate are not accounted for in the present model. The implication of using the average temperature is that a plate may have a fairly low average temperature, but a steep temperature gradient ranging from molten on one side to water saturation temperature on the other side. However, large deformations will not occur until plastic or high creep rate conditions exist in the entire plate thickness, which implies a high average temperature in the plate. Therefore, not considering the temperature gradient is not a serious drawback at the present level of detail in the mechanics models. The actual equations used are as follows:

$$M_c = \beta q b^2 / \epsilon \quad (4.13a)$$

$$M_e = -\frac{q b^2}{4\pi} \left(\ln\left(\frac{b}{c}\right) - \alpha \right) (1 + \nu_e) / \epsilon \quad (4.13b)$$

$$\sigma = \frac{6M}{h^2} \quad (4.14a)$$

$$\epsilon = \frac{d-a}{d} \quad (4.14b)$$

where

- M_c = the elastic moment at center of plate (N),
- M_e = the elastic moment at edge of plate (N),
- b = the length of plate edge (column pitch) (m),
- c = the diameter of support column (m),
- q = the load per unit area of plate (Pa),
- ν_e = the effective Poissons ratio,
- α = a constant = 0.811 for square plate cell,
- β = a constant = 0.0331 for square plate cell,
- ϵ = the ligament efficiency,

d = the hole pitch (m),
 M = either moment (N),
 σ = the peak stress at either location (Pa),
 h = the plate thickness (m), and
 a = the hole diameter (m).

The load q in the above equations is related to the actual load Q by the formula

$$q = Q \frac{d^2}{d^2 - \frac{\pi}{4} a^2} \quad (4.15)$$

where Q is the load per unit cell area.

The maximum plastic moment is determined by setting the maximum stress in the usual elastic equation equal to the yield stress and multiplying by the ratio between the maximum plastic moment and the moment at incipient yield (1.5):

$$M_p = 1.5\sigma_Y \frac{h^2}{6} \quad (4.16)$$

where M_p = the plastic moment of the plate, and
 σ_Y = the yield stress.

4.10.2 Elliptic Plate

The elliptic plate model describes an elliptic perforated shell, with allowance made for the possible presence of a central hole, as in the Westinghouse lower core support plate. This model is to be used for structures such as the vessel top, vessel bottom, or core support plate. The equations used are those for an elliptic shell [4.3] with the presence of perforations taken into account by modifying the mechanical properties. The central hole, if present, is modeled by applying a stress concentration factor. The model is used to evaluate the longitudinal and azimuthal stresses at the outside of each radial ring containing the plate. The shear stress at the crown is also evaluated if there is no central hole. The stress used in the LMP criterion is twice either the edge shear stress or the crown shear stress, whichever is greater; the stress applied to the failure criterion is limited to the yield stress, and no correction is made for stress distributions through the plate thickness.

The applied load is the average plate load corrected for the holes, as in Eq.(4.16) above. If no holes are present, then the

pressure differential across the two surfaces is also applied as a load. The elliptic shell equations are

$$\sigma_1 = Q' \frac{r_2}{2h}, \quad (4.17a)$$

$$\sigma_\theta = Q' \frac{r_2}{h} \left[1 - \frac{r_2}{2r_1} \right], \quad (4.17b)$$

$$Q' = P + q, \quad (4.17c)$$

$$r_2 = \left[(A^2 - R_o^2) \left(\frac{A}{B} \right)^2 + R_o^2 \right]^{1/2}, \quad (4.18a)$$

$$r_1 = r_2^3 \frac{B^2}{A^4}, \quad (4.18b)$$

where

- σ_1 = the longitudinal stress at outer edge of radial ring,
- σ_θ = the azimuthal stress at same location,
- A = the radius of plate major axis (radial),
- B = the radius of plate minor axis (vertical),
- R_o = the outer radius of radial ring,
- P = the pressure, and
- Q' = the total load.

The shear stress at R_o is then given by

$$\tau_o = \frac{\sigma_1 - \sigma_\theta}{2}, \quad (4.19)$$

where τ_o is the shear stress at the edge of the radial ring. The equations for the crown (with no central hole) are

$$\sigma_r = \frac{q}{2}, \quad (4.20a)$$

$$\tau_c = \frac{\sigma_1 - \sigma_r}{2}, \quad (4.20b)$$

where

- σ_r = the average radial stress, and
- τ_c = the shear stress at crown of shell.

If a central hole is present, then there is no crown and the edge shear stress for the edge of the innermost radial ring is multiplied by a stress concentration factor given as

$$K = 1 + \left(\frac{C}{R_o}\right)^2 \quad (4.21)$$

where

K = the stress concentration factor, and
C = the radius of central hole.

The above equations can also be used for a flat plate supported at the edges by letting the minor axis B approach zero. Use of the shell equations for thick plates such as the core support plate is somewhat dubious, since it neglects bending moments. However, since its application is at high-temperature, plastic conditions, this may not be too serious a limitation. The level of detail is commensurate with the level of the other mechanics models.

4.10.3 Cylinder

The thick wall equations for a cylinder are used to evaluate the stresses on the inner surface (the peak stress location for the hoop stress) [4.3]. The average axial stress is also determined. These stresses are then used to form the Tresca shear stress; twice the shear stress is used as input to the LMP criterion. The thick wall equations are

$$\sigma_r = qR_i^2 \frac{1}{h^2} \left(1 - \left(\frac{R_o}{R_i}\right)^2 \right), \quad (4.22a)$$

$$\sigma_\theta = qR_i^2 \frac{1}{h^2} \left(1 + \left(\frac{R_o}{R_i}\right)^2 \right), \quad (4.22b)$$

$$h^2 = R_o^2 - R_i^2, \quad (4.22c)$$

$$\sigma_z = \frac{q_z}{\pi h^2}, \quad (4.22d)$$

where

σ_r = the radial stress at inner wall,
 σ_θ = the azimuthal stress at inner wall,
 σ_z = the axial stress in the wall,
 q = the radial load,
 q_z = the axial load,

R_o = the radius of outer wall, and
 R_i = the radius of inner wall.

The Tresca shear stress τ is formed from the maximum and minimum of the principal stresses:

$$\tau = \frac{\sigma_{\max} - \sigma_{\min}}{2}, \quad (4.23a)$$

$$\sigma_{\max} = \max (\sigma_r, \sigma_\theta, \sigma_z), \quad (4.23b)$$

$$\sigma_{\min} = \min (\sigma_r, \sigma_\theta, \sigma_z). \quad (4.23c)$$

4.10.4 Columns

The column model is currently set up for cylindrical columns loaded in either compression or tension. An extended Euler buckling model [4.6] is used for compressive loads:

$$P_{cr} = \frac{\pi^2 E_t I}{L^2}, \quad (4.24a)$$

$$I = \frac{\pi (c_o^4 - c_i^4)}{4}, \quad (4.24b)$$

$$\sigma = \frac{q_z}{\pi (c_o^2 - c_i^2)}, \quad (4.24c)$$

where

P_{cr} = the critical buckling load,
 E_t = the tangent modulus,
 L = the length of column,
 I = the moment of inertia of the column cross-section,
 c_o = the outer column radius,
 c_i = the inner column radius,
 q_z = the axial load, and
 σ = the axial stress.

If the axial load is tensile, σ from Eq.(4.24c) is compared to the yield stress rather than to the buckling stress. The present model is limited to downward loads because of the coding in the STRESS routine; this will be generalized in future updates of MOD1.

4.10.5 Bolts

It is intended that there eventually will be a model for head bolts; however, this has not yet been implemented in MOD1.

4.10.6 Joints

It was intended also to include a model for joints between cylinders and plates. The main purpose of such a joint model was to account for secondary stresses produced by differential dilation under internal pressure at the vessel-head junction. This model is not included in MOD1.

4.10.7 Eggcrate Grill

The eggcrate model is included to allow analysis of the reinforced grill reactor support plate used in Babcock & Wilcox PWRs. The model can also be used for the upper core support plate in Westinghouse PWRs. The equations used are for a plate supported at the edges, and the effect of the reinforcing grill is included by increasing the plate rigidity [4.3]:

$$D = \frac{E h^3}{12(1 - \nu^2)} + \frac{EI}{d} , \quad (4.25)$$

where

D = Effective plate rigidity,
I = Effective moment of inertia of reinforced plate,
d = Beam (grill) pitch, and
 ν = Poisson's ratio.

For a typical B&W grill, the first term in Eq.(4.25) is several orders of magnitude smaller than the second and can be dropped.

The effective moment of inertia is calculated about the centroid of the grill section. The centroid is calculated as

$$x_c = \frac{1}{A} \int x \, dA , \quad (4.26a)$$

which, for the grill section in Fig. 4.5, is equal to

$$x_c = \frac{1}{2(hd + lb)} \left\{ bl^2 + d \left[(l + h)^2 - h^2 \right] \right\} , \quad (4.26b)$$

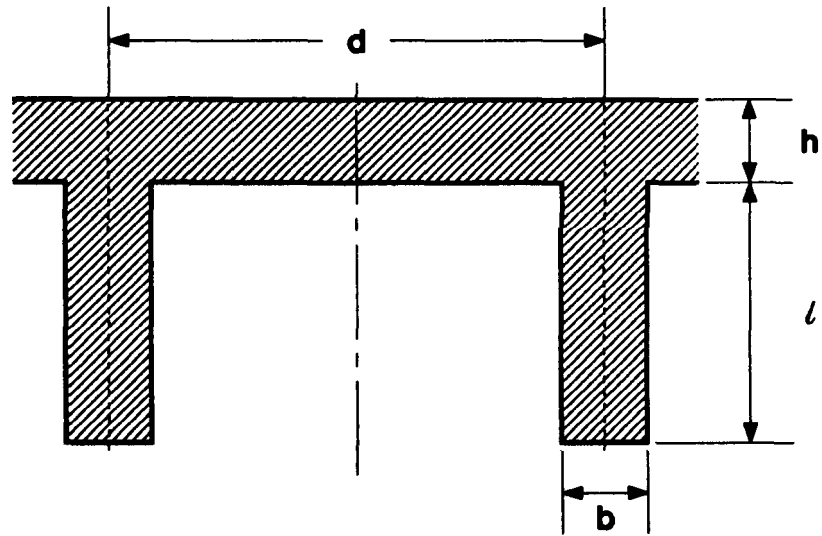


Figure 4.5 Eggcrate grill section.

where

x_c = centroid location relative to the bottom of the beam cross-section,
 h = plate thickness,
 b = beam width, and
 l = beam depth.

The moment of inertia is calculated by the usual formula

$$I = \int dy \int x'^2 dx' , \quad x' \equiv x - x_c . \quad (4.27a)$$

For the grill section under consideration, this equation reduces to

$$I = \frac{1}{3} \left[b \left((1-x_c)^3 + x_c^3 \right) + d \left((1+h-x_c)^3 - (1-x_c)^3 \right) \right] . \quad (4.27b)$$

The moment at the outer edge of a ring is calculated as

$$M_e = \frac{q}{16} (A^2(1+\nu) - R_o^2(1+\nu)) , \quad (4.28a)$$

where the parameters are the same as for the elliptic plate except for q , which is the same as for the grid plate (Eq.(4.13)). If no central hole is present, a central moment is also calculated as

$$M_c = \frac{q}{16} A^2(1+\nu) . \quad (4.28b)$$

The shear stress is then calculated as

$$\sigma_x = M_x \frac{1}{I} , \quad (4.29)$$

where the subscript x refers to the center or ring edge.

There are several points to note in reference to using the eggcrate model in MELPROG. The first is that the model refers to a plate and beams (grillwork) as a single structure; for a reasonable calculation, the properties in the above equations should be primarily characteristic of the beams rather than the plate; the properties are therefore not corrected for the presence of holes in the plate. Normally, this is not a problem, since most of the strength of the eggcrate comes from the beams, with the strength of the plate itself being a secondary

consideration. If it were important to consider the plate strength, it would be better to treat the grillwork and plate as separate structures, with the plate supported by the grillwork.

4.10.8 Example

Using data from the Zion FSAR [4.7], a cell for the lower grid plate is loaded by four fuel assemblies, resulting in a weight on the cell of 2600 kg. The approximate dimensions of the cell are as follows:

$$\begin{aligned}h &= 0.051 \text{ m,} \\a &= 0.074 \text{ m,} \\b &= 0.304 \text{ m,} \\c &= 0.086 \text{ m, and} \\d &= 0.152 \text{ m.}\end{aligned}$$

These numbers result in a ligament efficiency ϵ of 0.4, a load Q of 0.275 MPa, and an area ratio (cell area/actual area) of 1.23. The actual load per unit area q is then equal to $1.23Q$, or 0.34 MPa. The effective Poissons ratio for this case is 0.33, roughly the same as that of the actual material, 304SS, at low temperatures. At high temperatures near melting, it is 0.44. Using this high temperature value, we find

$$\begin{aligned}M_c &= 2600 \text{ N,} \\M_e &= -4100 \text{ N, and} \\M_p &= 0.0006\sigma_y.\end{aligned}$$

To put the above results in perspective, the yield stress σ_y of 304SS at 550 K (operating temperature for a PWR) is 130 MPa, resulting in a maximum plastic moment of 7.8×10^4 N. The grid plate is obviously under very little stress and will fail only at temperatures very close to the melting point. The yield stress at 1273 K has dropped to 20 MPa. However, it must drop to 4.8 MPa before the center of the plate cell is totally plastic. Based on a linear extrapolation from 20 MPa at 1273 K to 0 MPa at 1700 K, this stress corresponds to an average temperature about 100 K less than the melting point. Therefore, grid plate failure is most likely to occur only as its melting point is approached.

4.11 References

- 4.1 B. Carnahan, H. A. Luther, J.O. Wilkes, Applied Numerical Methods, John Wiley and Sons, New York, NY (1969).

- 4.2 D.V. Swenson and M.L. Corradini, Monte Carlo Analysis of LWR Steam Explosions, NUREG/CR-2307, SAND81-1092, Sandia National Laboratories, Albuquerque, NM (October 1981).
- 4.3 J.F. Harvey, Pressure Component Construction, Van Nostrand Reinhold (1980).
- 4.4 W.J. O'Donnell and B.F. Langer, "Design of Perforated Plates," Trans. of the ASME J. of Eng. for Industry, p. 307 (August 1962).
- 4.5 S. Timoshenko and S. Woinowsky-Krieger, Theory of Plates and Shells, McGraw-Hill (1959).
- 4.6 T.H. Lin, Theory of In-elastic Structures, John Wiley and Sons, Inc., New York, NY (1968).
- 4.7 Zion Nuclear Station, Final Safety Analysis Report, Commonwealth Edison Co. (1971).
- 4.8 J.S. Porowski and W.J. O'Donnell, "Elastic Design Methods for Perforated Plates," Trans. ASME, Vol 100, pp. 356-362 (April 1978).
- 4.9 A. Mendelson, PLASTICITY: Theory and Application, The MacMillan Company, New York, NY (1983).

5.0 RADIATION MODULE

A thermal-radiation heat transfer model for analyzing postulated LWR degraded core accidents must be able to treat a wide variation of core conditions. Among the specific core conditions that must be modeled are intact geometry, partially disrupted cores, rubble beds, cores with slumped fuel rods, molten pools, and various combinations of these. Also, the model must account for absorption and emission of thermal radiation in the water, vapor, and core debris that may exist in the reactor vessel.

To meet these modeling objectives, a thermal-radiation heat transfer model based on a combination of a net radiation enclosure model [5.1] and a flux model was developed. Each numerical cell is treated as a net radiation enclosure with internal structures and containing a homogeneous, nonscattering, absorbing, and emitting medium. The internal structures are modeled as surfaces that are diffuse emitters, absorbers, and reflectors. The components in the medium are treated as diffuse emitters and absorbers. The numerical cell boundaries are modeled as "black" surfaces that absorb all incident thermal radiation. The thermal radiation absorbed by the numerical cell boundary is treated as a directed flux that enters from the adjoining cell.

The cell boundary fluxes are the means by which cell-to-cell thermal radiation heat transfer is accomplished. The model treats each numerical cell as three-dimensional, that is three-dimensional view factors are used. The coupling to adjacent cells may be one-, two-, or three-dimensional. However, as implemented in MELPROG-PWR/MOD1 the coupling is two-dimensional (r,z).

5.1 Model Theory and Design

For an enclosure, the net radiative flux leaving surface k in the wavelength interval between λ and $\lambda + d\lambda$ is

$$dq_{\lambda,k} = dq_{\lambda o,k} - dq_{\lambda i,k} \quad (5.1)$$

where

$$dq_{\lambda o,k} = \begin{array}{l} \text{total radiative flux leaving surface } k \\ \text{in the wavelength interval between } \lambda \\ \text{and } \lambda + d\lambda \text{ and} \end{array}$$

$$dq_{\lambda i, k} = \begin{array}{l} \text{total radiative flux incident on surface } k \\ \text{in the wavelength interval between } \lambda \text{ and} \\ \lambda + d\lambda. \end{array}$$

Also, the total radiative flux leaving surface k in the wavelength interval between λ and $\lambda + d\lambda$ is

$$dq_{\lambda o, k} = \epsilon_{\lambda, k} e_{\lambda b, k} d\lambda + \rho_{\lambda, k} dq_{\lambda i, k} \quad (5.2)$$

where

$\epsilon_{\lambda, k}$ = emissivity of surface k at wavelength λ ,

$e_{\lambda b, k}$ = black-body emission term for surface k in the wavelength interval between λ and $\lambda + d\lambda$, and

$\rho_{\lambda, k}$ = reflectivity of surface k at wavelength λ .

Eqs.(5.1) and (5.2) can be combined to eliminate the $dq_{\lambda i, k}$ term. The resulting equation is

$$dq_{\lambda, k} = \frac{\epsilon_{\lambda, k}}{\rho_{\lambda, k}} \left(e_{\lambda b, k} d\lambda - \frac{1}{\epsilon_{\lambda, k}} (1 - \rho_{\lambda, k}) dq_{\lambda o, k} \right) \quad (5.3)$$

If the term $dq_{\lambda i, k}$ in Eq.(5.1) is expressed as the sum of the radiation leaving all other surfaces that is incident on surface k plus the radiation from the vapor that is incident on surface k , then Eq.(5.1) can be written as a function of $dq_{\lambda o, k}$:

$$dq_{\lambda, k} = dq_{\lambda o, k} - \sum_{j=1}^N \left(dq_{\lambda o, j} F_{k-j} \bar{\tau}_{\lambda, j-k} + e_{\lambda b, g} d\lambda F_{k-j} \bar{a}_{\lambda, j-k} \right) \quad (5.4)$$

where

F_{k-j} = view factor from surface k to surface j ,

$\bar{\tau}_{\lambda, j-k}$ = mean transmission coefficient for medium between surfaces j and k at wavelength λ ,

$e_{\lambda b, g}$ = black-body emission for medium between surfaces j and k in the wavelength interval between λ and $\lambda + d\lambda$,

$\bar{a}_{\lambda,j=k}$ = mean absorption coefficient for medium between surfaces j and k at wavelength λ , and

N = number of surfaces in the enclosure.

Combining Eqs.(5.3) and (5.4) to eliminate the $dq_{\lambda,k}$ terms results in a set of equations in which the dq_{λ_0} terms are the dependent variables. This set of equations is given by

$$dq_{\lambda_0,k} = \epsilon_{\lambda,k} e_{\lambda b,k} d\lambda + \rho_{\lambda,k} \sum_{j=1}^N (dq_{\lambda_0,j} F_{k-j} \bar{\tau}_{\lambda,j-k} + e_{\lambda b,g} d\lambda F_{k-j} \bar{a}_{\lambda,j-k}) \quad (5.5)$$

where k varies from 1 to N. The set of equations defined by Eq.(5.5) is linear and complete if the source terms $e_{\lambda b,k}$ and $e_{\lambda b,g}$ are known and if the view factors, mean absorption coefficients, mean transmission coefficients, emissivities, and reflectivities are constant. For these assumptions to be valid, the temperatures of the various surfaces within the enclosure must not change significantly during a time step.

In this model, all surfaces are assumed to be diffuse emitters, absorbers, and reflectors and to behave according to Kirchhoff's law. Therefore, the following relationships are valid:

$$A_k F_{k-j} = A_j F_{j-k} , \quad (5.6)$$

$$\rho_{\lambda,k} = 1.0 - \epsilon_{\lambda,k} , \quad (5.7)$$

$$\bar{a}_{\lambda,j-k} = \bar{a}_{\lambda,k-j} , \text{ and} \quad (5.8)$$

$$\bar{a}_{\lambda,j-k} = 1.0 - \bar{\tau}_{\lambda,j-k} . \quad (5.9)$$

The calculation of $\bar{a}_{\lambda,j-k}$ and $\bar{\tau}_{\lambda,j-k}$ depends on the intervening medium between surfaces j and k and on the view factor between surfaces j and k. Therefore, for a medium with a macroscopic absorption cross section, a_{λ} , for thermal radiation at wavelength λ , a transmission coefficient can be defined as

$$\tau_{\lambda}(s) = \exp(-a_{\lambda} s) \quad (5.10)$$

where s is the path length. The mean transmission coefficient is then defined to be

$$\bar{\tau}_{\lambda, j-k} = \frac{1}{A_j F_{j-k}} \int_{A_k} \int_{A_j} \frac{\tau_{\lambda}(s) \cos \beta_k \cos \beta_j}{\pi s^2} dA_j dA_k \quad (5.11)$$

where

$$F_{j-k} = \frac{1}{A_j} \int_{A_k} \int_{A_j} \frac{\cos \beta_k \cos \beta_j}{\pi s^2} dA_j dA_k \quad (5.12)$$

is the view factor between surfaces j and k and, where A_j and A_k are the areas of surfaces j and k , respectively. The view factors, mean transmission coefficients, and mean absorption coefficients are calculated as a function of time and space.

The numerical cell boundaries are modeled as "black" surfaces, that is they have emissivities of 1.0 and reflectivities of 0.0. Therefore, for the cell boundaries, Eq.(5.5) can be simplified to

$$dq_{\lambda o, k} = dq_{\lambda b, k} \quad (5.13)$$

where $dq_{\lambda b, k}$ is the flux entering the enclosure through the cell boundary from an adjoining cell. Application of Eq.(5.13) results in a reduction in the order of the matrix that must be solved for each cell by the number of cell boundaries. However, because $dq_{\lambda b, k}$ is an unknown, an iterative solution procedure must be used for the boundary fluxes.

Once the values of $dq_{\lambda o}$ have been determined, the net radiation heat transfer from the structure surfaces and from the medium can be determined. For surface k the net radiation heat transfer leaving it is given by

$$Q_k = A_k \int_{\lambda} (dq_{\lambda o, k} - dq_{\lambda i, k}) d\lambda \quad (5.14)$$

where

A_k = area of surface k , and

$$dq_{\lambda i, k} = \sum_{j=1}^N (dq_{\lambda o, j} F_{k-j} \bar{\tau}_{\lambda, j-k} + e_{\lambda b, g} d\lambda F_{k-j} \bar{a}_{\lambda, j-k}) \quad (5.15)$$

For the medium, the net radiation heat transfer leaving it is determined from the net radiation heat transfer leaving the

surfaces. The net radiation heat transfer from the medium is given by

$$Q_g = - \sum_{k=1}^N Q_k \quad . \quad (5.16)$$

5.2 Radiation Heat Transfer Model Implementation

5.2.1 Geometry and Fluid Fields Modeled

Two-dimensional numerical cells in an R-Z cylindrical mesh are treated by the radiation heat transfer model. Within a numerical cell, the MELPROG-PWR/MOD1 model can treat any combination of the following structures:

1. a wall at the cell inner radius,
2. a wall at the cell outer radius,
3. a plate at the cell bottom,
4. a plate at the cell top,
5. fuel rods,
6. control rods, and
7. no structures at all.

Additionally, the top and bottom surfaces of a debris bed are treated as structure surfaces if they are present in the cell. Water, steam, and corium are treated as a homogeneous mixture in cells without a debris bed, while only water and steam are treated as a homogeneous mixture in cells with a debris bed. The above list of structures does not include columns or "general" structures. Columns will be treated in future versions of MELPROG; however, because detailed geometry is not provided for "general" structures, they will not be treated in the radiation model. Other structures that might be added to MELPROG in the future will be considered for inclusion in the radiation heat transfer calculation at that time. Figure 5.1 shows typical geometries that are currently treated by the RADIATION module.

5.2.2 Solution Procedure

The radiation heat transfer solution is partitioned into several steps. The first step is to calculate black body emission terms for each structure surface and a mixture term for the fluid fields. Currently only one energy group is used. Therefore, these terms are σT^4 for each surface, while for the fluids, σT^4 is calculated for each field and then averaged using a weighting factor based on its macroscopic absorption cross

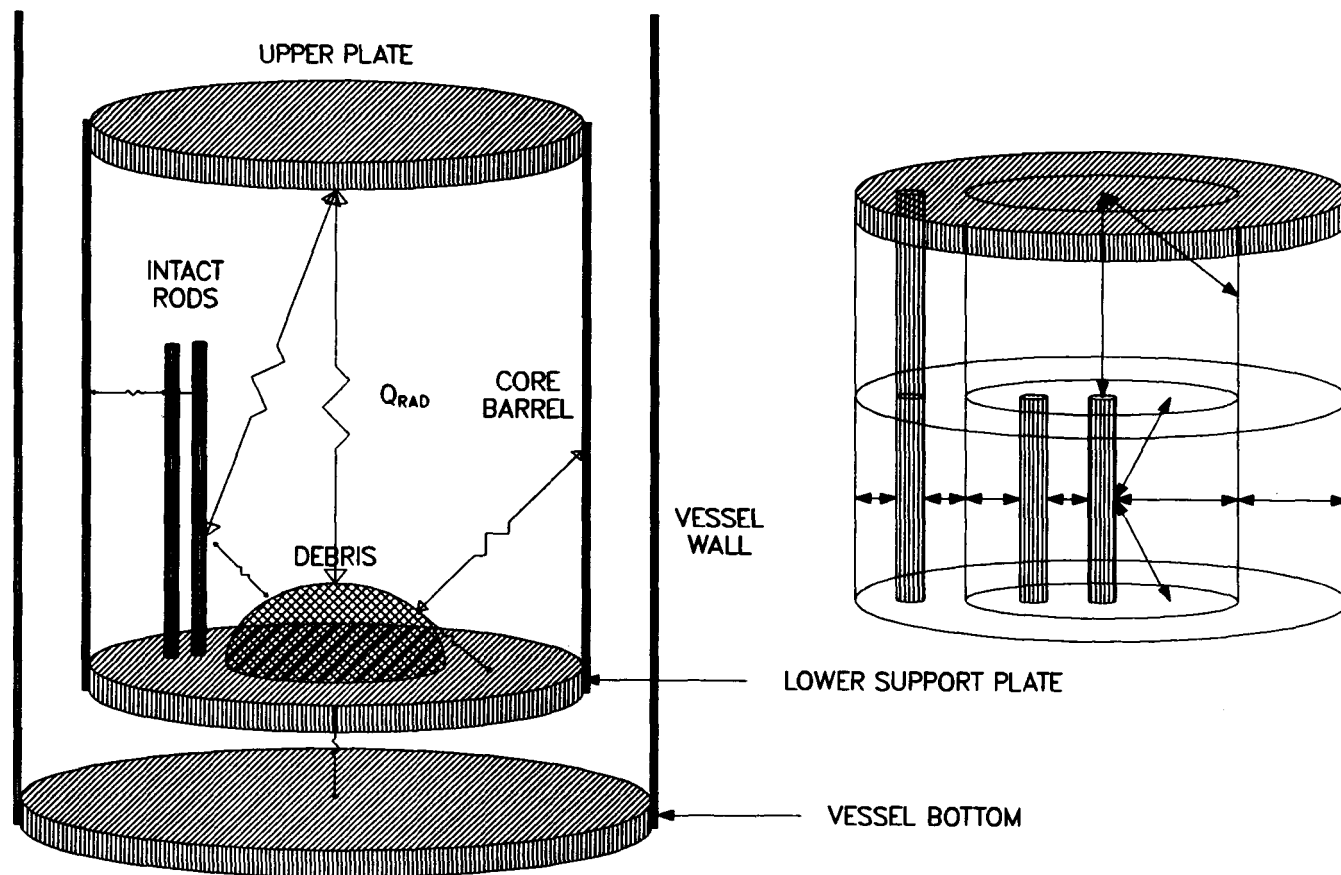


Figure 5.1 Typical Geometries Treated in RADIATION Module

section. Also, as part of the first step, the individual macroscopic absorption cross sections and a mixture absorption cross section are calculated.

In the second part of the solution procedure, view factors are calculated (if there have been geometric changes in a cell) and mean transmission coefficients are calculated. Details of the view factor calculation are presented in Section 5.2.3 and details of the mean transmission coefficient calculation are presented in Section 5.2.4.

Part three of the solution assembles the coefficient matrix and calculates a portion of the resultant vector for each cell. The resultant vector calculation is divided into parts because some of the terms that make it up change during the iteration performed in part five.

In part four, the coefficient matrix assembled for each cell in part three is inverted.

In part five, an iteration on the cell boundary fluxes is performed. The iteration is necessary because the radiative flux entering a cell from an adjoining cell depends on the solution for the fluxes in that adjoining cell just as the radiative flux calculated to be leaving a particular cell affects the solution in the cell that the flux is entering. The first step in the iteration is to calculate the changing part (the part affected by cell boundary fluxes) of the resultant vector for each cell. The solution vector for a cell is then calculated by multiplying the inverted matrix by the resultant vector. The new cell boundary fluxes (i.e., fluxes that are leaving the cell) are calculated and a check on convergence is made between iterations. (For the first iteration the fluxes from the previous time step are used for the convergence check.) The iteration continues until all boundary fluxes have met a user input convergence criterion (usually about 10^{-5}) or a user input iteration limit has been reached (usually about 50). Convergence is normally achieved after four or five iterations; however, when there have been significant geometry changes or the individual cells are tightly coupled many more iterations may be required.

The final part of the radiation heat transfer solution is the calculation of net radiative fluxes to each structure surface and to each fluid field. These calculations are performed for each two-dimensional numerical cell.

5.2.3 View Factor Calculation

As stated in Section 5.2.1, not all possible structures are treated by the RADIATION module in MELPROG-PWR/MOD1. The view factor calculation can be extremely complicated even for relatively simple geometries. To simplify this problem somewhat, certain structures have been neglected (currently columns and "general" structures) and some assumptions about the location of other structures in the two-dimensional mesh have been made. Walls are always assumed to coincide with either the inner or outer radius of a cell. Plates are always assumed to coincide with either the top or bottom boundary of a cell. In cells with fuel rods and/or control rods, it is assumed that there are a sufficient number of rods so that there is no line of sight path through the rods in the radial direction.

In MELPROG-PWR/MOD1 the view factor calculation has been organized to treat two types of cells: cells without fuel rods or control rods (type 1) and cells with fuel rods and/or control rods (type 2). Within the assumptions made about structure locations and the treatment of radiation heat transfer across cell boundaries, all the view factors for type 1 cells can be calculated using analytical solutions; however, for type 2 cells some approximations must be made. There are three analytical view factor functions used in MELPROG-PWR/MOD1. These include a cylinder's inner surface to an internal cylinder's outer surface, a cylinder's inner surface to itself shadowed by an internal cylinder, and a disk to a parallel disk of equal diameter. View factors between surface 2 (see Fig. 4.2) and its adjacent cell boundary are defined to be 1.0. Therefore, surface 2 of a structure always radiates to an adjoining cell except when it is on a problem boundary. Walls and plates (including plates with holes) are assumed to be opaque to all thermal radiation. Plates with holes will be properly treated in future code versions. The thickness of walls, plates, and debris beds are accounted for in the calculation of view factors, for example, as a debris bed increases in depth in a cell the view factors in that cell will change.

For a type 1 cell, only two view factors must be explicitly calculated. These are a cylinder to an internal cylinder and a cylinder to itself shadowed by an internal cylinder. All of the other view factors that are needed for a type 1 cell are determined by applying Kirchhoff's law (Eq.(5.6)), by requiring

that each view factor be equal to or greater than zero, and by requiring that the sum of all view factors from any given surface be equal to 1.0. Table 5.1 shows how the view factor matrix would look for a cell with an outer wall and a bottom plate (for example a cell with the core barrel and diffuser plate). If this sample cell had a debris bed in it in addition to the other surfaces, the view factor calculation would be performed in the same way except that all of the view factors to surface 1 of the bottom plate would be to the debris bed surface. The view factor for surface 1 of the bottom plate to itself would be defined to be 1.0.

In a type 2 cell (cells with fuel rods and/or control rods), there are some approximations that must be made to utilize the three analytical view factor solutions discussed above. First, the fuel rods are assumed to approximate a annulus whose outer diameter is one-half the fuel rod pitch smaller than the outer cell radius or the inner radius of an outer wall and whose inner diameter is one-half the fuel rod pitch larger than the cell inner radius or the outer radius of an inner wall. Second, the fuel rods and/or control rods in a cell are assumed to be of sufficient number that there is no line of sight through them in the radial direction. Third, the view factor from the cell bottom to the cell top through the rods is approximated by the view factor from a disk, having an area equal to the total cell bottom surface area divided by the number of rods in the cell minus the area of one rod, to a disk having the same area at the cell top. By using these three approximations, together with Kirchhoff's law (Eq.(5.6)), the requirement that each view factor be equal to or greater than zero, and the requirement that the sum of all view factors for a surface be 1.0, a consistent set of view factors for type 2 cells are calculated. This procedure is valid for PWR whole core calculations; however, for BWRs or small bundle experiments the approximations are invalid. A more detailed numerical view factor solution is required to adequately treat BWRs and small fuel rod bundles.

When both fuel rods and control rods are present in a type 2 cell, the fuel rod view factors calculated using the above procedure must be modified. The procedure used to modify the fuel rod view factors is based on two assumptions. The first assumption is that fuel rods and control rods are homogeneously mixed and the second assumption is that control rods and fuel rods have approximately equal diameters. The first assumption implies that the fuel rod to fuel rod and fuel rod to control rod view factors sum to equal the original fuel rod to fuel rod view factor. And, the second assumption leads to view factors from

Table 5.1 View Factor Matrix for a Typical Type 1 Cell

		Outer Wall		Bottom Plate		Boundary			
		Surface 1	Surface 2	Surface 1	Surface 2	Bottom	Top	Inner Radial	Outer Radial
Outer Wall	Surf. 1	$F_{W1,W1}$	0.0	$F_{W1,P1}$	0.0	0.0	$F_{W1,TB}$	$F_{W1,IRB}$	0.0
	Surf. 2	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0
Bottom Plate	Surf. 1	$F_{P1,W1}$	0.0	$F_{P1,P1}$	0.0	0.0	$F_{P1,TB}$	$F_{P1,IRB}$	0.0
	Surf. 2	0.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0
Boundary	Bottom	0.0	0.0	0.0	1.0	0.0	0.0	0.0	0.0
	Top	$F_{TB,W1}$	0.0	$F_{TB,P1}$	0.0	0.0	0.0	$F_{TB,IRB}$	0.0
	Inner Radial	$F_{IRB,W1}$	0.0	$F_{IRB,P1}$	0.0	0.0	$F_{IRB,TB}$	0.0	0.0
	Outer Radial	0.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0

NOTE: In this Table "F" means view factor and the subscripts have the following meanings:

W1 means wall surface 1,
P1 means plate surface 1,
TB means top boundary, and
IRB means inner radial boundary.

fuel rods to the bottom, top, inside, and outside being equal to view factors from control rods to the bottom, top, inside, and outside, respectively. If the original fuel rod to fuel rod view factor is F_{pp} , then

$$F_{pp} = F'_{pp} + F_{pc} \quad (5.17)$$

and

$$F'_{pp} = \frac{A_p}{A_c + A_p} F_{pp} \quad (5.18)$$

where

F'_{pp} = fuel rod to fuel rod view factor considering control rods,

F_{pc} = fuel rod to control rod view factor,

A_p = total fuel rod surface area (m), and

A_c = total control rod surface area (m).

The view factor calculation in MELPROG-PWR/MOD1 is dynamic; that is, whenever there are geometry changes in a cell the view factors for that cell are recalculated. Examples of situations when view factors are recalculated are when a structure melts, when fuel rods balloon or disintegrate, and when debris beds form or change in thickness.

5.2.4 Material Properties and Mean Transmission Coefficients

The RADIATION module uses constant emissivities for each structure surface and for debris bed surfaces. The emissivities are user input constants. The water and corium mass absorption cross sections, which are user input constants, are also required. For steam, a temperature and pressure dependent function is used to calculate a Planck mean absorption cross section. Hydrogen is treated as transparent to thermal radiation. In future versions of MELPROG, temperature-dependent emissivities will be provided. The current version treats only one energy group; however, the model is designed to treat multiple energy groups and could do so with additional emissivity and absorption cross section data and this option may be included in future versions.

The calculation of mean transmission coefficients using Eq.(5.11) is extremely difficult and cannot be done analytically for the geometries considered in the RADIATION module.

Therefore, Eq.(5.10) is used with an average path length(s). The path length calculation is dependent on the cell geometry and varies as the cell geometry changes during an accident calculation.

5.2.5 Summary of Basic Modeling Assumptions

It is assumed in the RADIATION module that all surfaces are diffuse emitters, absorbers, and reflectors; that the fluid fields are homogeneously mixed; and that radiation across numerical cell boundaries is diffuse. Furthermore, it is assumed that it is not necessary to treat every surface, that fuel and control rods are always vertical and in sufficient numbers to prevent horizontal line-of-sight viewing through them, that there are no holes in plates or walls, that all plates are flat, and that all walls are cylindrical. Also, it is assumed that all temperatures are constant within a time step and, therefore, do not change substantially between time steps.

5.2.6 RADIATION Module Interaction with Other Modules

The RADIATION module interacts with each of the other calculation modules in MELPROG. From the FLUIDS module, it gets densities and temperatures and it returns volumetric energy transfer rates. From the CORE module, it gets fuel and control rod temperatures and dimensions and it returns radiative fluxes. From the STRUCTURES module, it gets geometry and temperatures and it returns radiative fluxes. Finally, from the DEBRIS module, it gets debris bed geometry and temperatures and it returns radiative fluxes.

5.3 References

- 5.1 R. Siegel and J.R. Howell, Thermal Radiation Heat Transfer, McGraw-Hill Book Company, pp. 550-558 (1972).

6.0 DEBRIS MODULE

Rubble beds that form in the reactor core and the lower plenum are identified and analyzed in the DEBRIS module. The DEBRIS module employed in the MELPROG/MODO code is currently used in the MOD1 version of the code: this original version of the module is described in Sections 6.0 through 6.3. Recently, a mechanistic, two-dimensional debris meltdown model has been developed and is being implemented in the MELPROG code: a detailed description of this work is provided in Section 6.4.

In this report the term "debris" represents all forms of disintegrated fuel rods, control rods, and structural components. Three states of the debris are considered in the modeling: (1) stationary, pre-dryout regime that occurs when debris accumulates on grid spacers or freezes and forms a blockage between intact fuel rods when liquid water is still present; (2) stationary, post-dryout regime in which no liquid water is present to cool the debris and the temperature can reach the uranium melting point; (3) nonstationary regime that occurs when a structure supporting a debris bed fails. The second case is modeled in the DEBRIS module while calculations for the first and third regimes are performed in the FLUIDS module.

The DEBRIS module is comprised of two main parts, a bed formation and release model and a post-dryout debris analyzer. Dry, stationary debris beds are identified using the formation model. Boundary and initial conditions are then determined and the post-dryout model is activated. The transient temperature distribution in the bed is determined by considering heat sources due to decay heat and oxidation as well as conduction, convection, and radiation heat transfer in the debris. The model also calculates the melting of debris and relocation of the melt within the bed. Simplified phase diagrams are used to fix the state of the debris and the relocation step uses simplified laws for porous flow to predict the behavior of the molten material. From this analysis, mass and composition profiles in the debris are calculated as a function of time. The DEBRIS module also provides boundary conditions for the STRUCTURES heat transfer model and input to the RADIATION module.

A typical calculation sequence is traced in the following section. The bed formation and post-dryout models are described in Sections 6.2 and 6.3, respectively. Particular attention is given to assumptions made in the analysis.

6.1 Calculation Sequence

If there is no debris present in the problem, then this module is not active. However, once debris is formed, the DEBRIS module is activated and the debris bed formation and release model is entered. This model is responsible for

1. determining the location of debris beds,
2. defining the state of the debris beds, and
3. determining if the debris beds should release.

The model examines the FLUIDS data arrays to determine the location of debris throughout the vessel. This information is used to determine whether or not a detailed calculation should be performed. The debris must be dry and stationary before a detailed calculation is initiated. If it is determined that a dry debris bed has formed, then this model sets a flag to begin a detailed calculation and determines the initial and boundary conditions for the bed. Otherwise, no flag is set and the behavior of the debris is performed by FLUIDS.

If the formation model determines that a detailed calculation should not be performed, then the debris is analyzed in either the FLUIDS or IFCI modules. If water is present and the debris is stationary, temperatures in the bed are low, temperature gradients are small and the coarse-grid calculation performed in the FLUIDS module is adequate. When molten debris is moving, its behavior will be calculated in the IFCI module (once this module is fully implemented).

If a dry debris bed is formed, then the post-dryout debris bed model calculates the debris behavior. This model combines a mixture energy equation for the debris bed with equations for melt relocation. One-dimensional geometry (axial) is used for simplicity. The energy equation is time dependent and includes internal heat generation (both chemical oxidation and decay heat), conduction, convection and radiation heat transfer.

If a detailed calculation is performed, the following steps are taken. At each time step, a mixture energy equation is solved. This equation is written in terms of an equilibrium temperature and, by solving this equation, the temperature distribution within the bed can be calculated. In this model, it is assumed that the melt and solid are in equilibrium. This

assumption is reasonable in view of the large surface area of the debris and tightly packed nature of the debris bed which result in high flow resistance and concomitant low velocities. It is also assumed that the radiation heat transfer within the bed can be accounted for by using an effective conductivity model: based on several in-pile melt progression experiments, this latter assumption is quite reasonable [6.1].

After the energy equation is solved, the melt formation and relocation part of the model is entered. Using simplified phase diagrams, the state of the debris (i.e., its composition and temperature) and the amount of liquid (if any) within a bed can be calculated. If liquid exists, it is relocated using a semi-empirical model. Following relocation, the melt within the bed is equilibrated with the solid which may cause freezing of the melt. The porosity distribution along with the composition distribution are also updated by again referencing the phase diagram. In the current version, debris materials can include UO_2 , Zr, ZrO_2 , $Zr(O)-UO_2$, stainless steel, and the Ag-In-Cd alloy control rod material.

After the relocation step, material properties for the debris are recalculated. This marks the end of a time step. At this point new temperatures and porosities have been calculated for the bed. This final result is then used as the initial condition for the next time step.

During each time step of the detailed calculation, the debris bed formation and release model performs three additional functions. First, it determines whether debris is being added to the top of the bed or added internally (e.g., from rod break-up) and updates the geometry accordingly. Second, the boundary conditions for the bed are updated based on FLUIDS, RADIATION and STRUCTURES information. Finally, the debris release part of the model determines if the bed supporting structure has failed. If this occurs, the detailed calculation ceases and the bed is allowed to relocate using the FLUIDS module.

The above discussion has simply traced through the calculational sequence. The specific details of the models are described in the following sections.

6.2 Debris Bed Formation and Release Model

The debris bed formation and release model is concerned with identifying the locations of debris beds, determining their initial and boundary conditions, and determining when they should release. In the model, the area of greatest uncertainty lies in determining how the debris progresses from its initial formation stage to the point where it can be called a debris bed. Certainly, this is one area where experimental activities are crucially needed to improve the current state of knowledge.

Debris forms if fuel rods, control rods and structural components disintegrate. Once this debris forms, the mass and energy of the fuel rods or control rods as well as their effective diameter (or surface-to-volume ratio) are supplied to the FLUIDS module. FLUIDS then begins to transport debris and calculate the heat transfer between the debris and the coolant. In general, the debris will cool and move downward.

At each time step, the debris formation model examines the FLUIDS data arrays to determine if a debris bed has formed. There are three criteria that can lead to a bed being formed. The first criterion is the most obvious and states that if the debris accumulates on an impermeable plate, then a debris bed will begin to form. The second criterion is similar and states that if the cell below a given cell is full of debris, then the upper cell can also be considered to be a debris bed cell. The third criterion is based on the local debris fraction exceeding a specified limit. Currently, debris beds are allowed to form in any cell in which the local debris fraction exceeds limits as defined by Moore [6.2]. These limits are based on experimental and numerical studies of debris packing fractions for Three-Mile Island type geometries. Typically, the packing fraction is approximately 0.44. However, if the debris is entirely molten, a higher limit of 0.9 is used.

The application of the first and second criteria are straightforward, but the application of the third criterion requires the calculation of the local debris fraction in each cell (i.e., the two-dimensional distribution). The local debris fractions are calculated in the DEBRIS module by solving auxiliary continuity equations. In solving these equations, it is assumed that the debris velocity is radially uniform and that there is no radial motion of the debris.

The formation model uses these three criteria to identify the locations of debris beds in the vessel. If a cell contains a bed, then the debris velocity at the bottom of that cell is set to zero.

Once a bed has formed in a cell, the model examines the conditions of the debris to determine whether or not it is dry and static. For pre-dryout or non-stationary conditions, the heat transfer and momentum exchange between the debris and the liquid and vapor are calculated exclusively by FLUIDS. If the debris is dry and static, then the bed formation model obtains the initial temperature, porosity, and composition of the bed based on values obtained from FLUIDS. After these conditions have been obtained, the model nodalizes the bed according to user specification. The boundary conditions for the bed are obtained by searching STRUCTURES and RADIATION data arrays to determine either an appropriate boundary temperature or surface heat flux. The vapor flow rate through the bed is obtained from FLUIDS.

The initial conditions for a bed are only obtained once. However, on subsequent time steps, the boundary conditions and vapor flow rate through the bed may vary. Consequently, these quantities are continually updated.

The model accounts for changes in the geometry of the debris bed. Specifically, if debris is added to the top of the bed, the model allows for the bed height to grow with time by adding nodes to the bed as necessary. Also, if beds form among intact rods and the rods subsequently fail, the debris is added internally to the bed by adjusting the composition, flow area, and porosity of the bed.

In addition to forming beds and updating their geometry, the model also determines when beds should release. The word "release" means that the beds are no longer supported and, hence, can start moving. Obviously, if the structure which is supporting a debris bed fails, then the bed will begin moving. This represents a transition from the stationary regime to the non-stationary regime so that detailed calculations cease. Of course, when the debris again comes to rest, it may form a new bed. However, the detailed information about the bed behavior is not retained during this transition.

The release of a bed is controlled by examining the STRUCTURES and CORE data arrays on each time step to determine if

the bed supporting structures have failed. STRUCTURES provides information concerning the failure of support plates which would be one mechanism for bed release. If a bed is resting on a structure which fails, then the bed will release.

6.3 Post-Dryout Debris Bed Model

This section discusses the details of the post-dryout debris bed model. Both the heat-up part as well as the melt relocation part are discussed. Additional modeling aspects of the problem are also described.

6.3.1 Heat Transfer

As discussed above, a mixture energy equation is solved in the model to determine the temperature distribution in the debris. Assuming thermal equilibrium between the solid and the gas gives

$$\left[(1-\epsilon)\rho C_p \right]_s \partial T / \partial t = V \cdot k_{eff} \nabla T + Q - h a_v (T - T_g) \quad , \quad (6.1)$$

where $(\rho C_p)_s$ is the effective volumetric specific heat of the solid (and/or liquid) and is given by

$$(\rho C_p)_s = \sum (\alpha \rho C_p)_i \quad (6.2)$$

where the summation is over all components of the bed and ϵ is porosity, ρ is density (kg/m^3), c_p is specific heat (J/kg-K), T is temperature (K), t is time (s), k_{eff} is the effective conductivity (W/m-K), Q is volumetric power (W/m^3), α is the component volume fraction, h is a heat transfer coefficient, a_v is the surface area of the debris per unit volume and T_g is the temperature of the gas. The term on the left hand side represents the rate of change of the mixture energy (both solid and gas). The first term on the right hand side represents the energy gained by conduction. An effective conductivity is used to account for both conduction and radiation heat transfer effects. The next term on the right hand side represents the energy gained by internal heat generation and the last term accounts for heat transfer to the gas.

A standard finite difference approximation for the differential equation is obtained by integrating the equation over the volume of a node. The temperature of a node is defined at the node center and represents the space-averaged temperature

of the node. In the following equations the center of a node has axial index j . Integrating over the volume of a node and rearranging gives:

$$AZ \left[((1-\epsilon)\rho C_p)_s \frac{\partial T}{\partial t} - Q \right] - A \left[k_{\text{eff}} \frac{\partial T}{\partial z} \right] \Big|_{j-1/2}^{j+1/2} + AZ h a_v (T - T_g) = 0 \quad (6.3)$$

where A is the area (m^2) and Z is given by

$$Z = z_{j+1/2} - z_{j-1/2} \quad (6.4)$$

This equation represents the volume average energy equation for a node in the problem. Standard differencing approximations are used to arrive at the finite difference form of the energy equation which is given by

$$AZ [((1-\epsilon)\rho c_p)_s (T - T^n)/\Delta t - Q] - A \left[(k/\Delta z)_{j+1/2} (T_{j+1} - T_j) - (k/\Delta z)_{j-1/2} (T_j - T_{j-1}) \right] + AZ h a_v (T - T_g) = 0 \quad (6.5)$$

where

$$(k/\Delta z)_{j+1/2} = \frac{k_j k_{j+1} (\Delta z_j + \Delta z_{j+1})}{k_j \Delta z_{j+1} + k_{j+1} \Delta z_j} \quad (6.6)$$

$$\Delta z_j = z_{j+1/2} - z_{j-1/2} \quad (6.7)$$

and

$$\Delta z_{j+1/2} = z_{j+1} - z_j \quad (6.8)$$

The temperature, T , is treated in a fully implicit sense in that all temperatures except T^n are evaluated at the new time step $(n+1)$.

Most of the material properties are allowed to be temperature dependent and must be continually updated during a calculation. Additionally, the conductivity of the debris depends on the porosity and volume fraction of the constituent materials. Two debris conductivity models are included in the code, both of which have been described in Reference 6.1. The first is the Imura-Takegoshi/Vortmeyer model [6.3,6.4] which has been

demonstrated to represent an upper bound conductivity. The second model is the Willhite-Kunii-Smith/Luikov model [6.5,6.6] that yields a lower bound conductivity. Both models include the effects of conduction and radiation within the bed. The multi-component nature of the debris is incorporated using the Maxwell model [6.6].

6.3.2 Melt Relocation Model

Melt relocates in a rubble bed under the influence of gravity, capillary and drag forces. In the current version of DEBRIS, both gravity and drag forces are neglected and the effect of capillary forces is included using the concept of a melt penetration distance. When melt forms the local porosity increases and the liquid is drawn out of the melt zone by capillary forces. The liquid is assumed to relocate instantaneously and the maximum penetration is estimated from both hydrodynamic and thermal (freezing) considerations.

Assuming that the motion of the liquid ceases when it has entirely left the melt zone, the maximum penetration is

$$X_{\max} = \frac{X_o \epsilon_r}{\epsilon_b} \quad (6.9)$$

where X_o is the volume of the liquid divided by the cross-sectional area, ϵ_r is the porosity of the melt zone and ϵ_b is the bed porosity. If the melt is completely deposited as a film on the particles,

$$X_{\max} = \frac{X_o \epsilon_r}{\epsilon_b \gamma} \quad (6.10)$$

where γ is the fraction of the flow area which contains the film. Values of the film thickness as a fraction of the bed porosity range from 0.75 for a bed of 10% porosity to 0.12 for a bed of 60% porosity [6.7]. Hence, film deposition can reduce the distance the melt can relocate.

The above discussion has only considered hydrodynamic effects. In an actual bed, there will be temperature gradients that will allow the melt to freeze and possibly form blockages in the bed. The hydrodynamic limits may be viewed then to be absolute maximum limits since thermal effects will shorten the

penetration distance. In view of these potential thermal effects, a third criterion has been derived to estimate the penetration distance.

A simplified steady-state heat balance for molten debris that is freezing is given as

$$(1-\epsilon)(\rho C_p)_s (T_m - T_i) = \epsilon \rho_l h_{ls} \quad (6.11)$$

where T_m is the melt temperature, T_i is the local bed temperature, and h_{ls} is the heat of fusion. This equation can be rearranged to find the maximum bed temperature, T_{max} , that will freeze all the adjacent liquid:

$$T_{max} = T_m - \frac{\epsilon \rho_l h_{ls}}{(1-\epsilon)(\rho C_p)_s} \quad (6.12)$$

If the bed temperature is less than T_{max} , then complete freezing is possible. If the heat transfer rate is very large, then this temperature limit can be used to estimate the position where complete freezing and, hence, maximum penetration will occur. Due to the large surface area in the bed, the heat transfer is expected to be large. Hence, with the calculated temperature distribution and this temperature limit, the maximum penetration distance based on thermal effects can be calculated.

With these simple equations for the melt penetration distance, it is now possible to describe how the melt relocation model works. The calculational sequence in this model begins after the temperature distributions have been calculated. With the calculated temperature distribution, the amount of liquid (if any) is calculated in the phase diagram routine. If liquid is present, the melt relocation routine is entered.

There are two possible paths for calculating the relocation depending on whether or not a stable flow blockage has formed. If no blockage has formed, then the size of the melting zone is determined and the mass and energy of the molten material is calculated. The upward penetration distance from the top of molten zone is calculated and an attempt is made to move 50% of the melt this distance (it is assumed that capillary forces act in a symmetric manner). The porosity distribution is updated and if a blockage occurs, then the excess melt is reserved for downward flow.

Following the upward relocation, the downward penetration distance from the bottom of the molten zone is calculated. An attempt is then made to move the remaining melt this distance. Again porosities are updated and if a blockage occurs, the melt is backfilled on top of the blockage.

Following the relocation step, the temperatures of the liquid and solid in each node are equilibrated. The equilibration is assumed to occur instantaneously which is reasonable in view of the large surface area in the bed. Once equilibrated, the phase diagram routine is used to define the state of the bed.

The relocation model also has a few additional features which are worth noting. First, there is an upper limit on the porosity of a node. If the porosity exceeds this limit, then the node is assumed to collapse down into the node below. Second, mass conservation equations are solved for all components of the bed. This means that the distribution of UO_2 , ZrO_2 , Zr, $\text{Zr(O)}\text{-UO}_2$, steel and control rod material within the bed are calculated.

6.3.3 Phase Diagrams

The phase diagram routine in the DEBRIS module uses the temperature and composition of the debris to predict the state of the debris. The model is relatively simple, relying on simplified ternary and binary phase diagrams. The U-Zr-O system is fairly complicated, but the limited knowledge in this area has been combined to produce a simple but functional method of determining the state of the debris, and in particular the melting and freezing of the debris constituents.

In the following discussion the term "melting" is used to mean liquid formation and the term "freezing" is used to mean solid formation. The current model considers melting and/or freezing of the following constituents:

1. Ag-In-Cd control rod alloy,
2. stainless steel,
3. Zr metal,
4. UO_2 ,
5. ZrO_2 ,
6. $\text{UO}_2\text{-ZrO}_2$, and
7. $\text{Zr(O)}\text{-UO}_2$.

The first three constituents are treated as simple single phase melting and freezing processes. Although the steel and control

rod material are alloys, they are assumed to form ideal liquid and solid solutions. The $\text{UO}_2\text{-ZrO}_2$ system is treated as a simple binary eutectic that forms a liquid with a UO_2 concentration of 50 mole percent. This liquid will form if UO_2 and ZrO_2 are the only solids present in a given node and the temperature is high enough. The Zr(O)-UO_2 system is treated as a binary system in which the composition of the liquid is temperature dependent. Liquid is formed by first melting the Zr metal and then dissolving the UO_2 into it. The amount of UO_2 dissolved will depend on the local temperature.

Table 6.1 summarizes the phase rules used in the model. If the composition and temperature of a node is given, then the amount of solid and liquid in the node can be calculated.

Table 6.1		
Summary of DEBRIS Phase Change Rules		
<u>Component</u>	<u>Melting Temperature</u>	<u>Type of Melting</u>
Ag-In-Cd	1070 K	Single Component
Steel	1700 K	Single Component
Zr Metal	2200 K	Single Component
Zr(O)-UO_2	2200-2650 K	UO_2 Dissolves into Molten Zr
$\text{ZrO}_2\text{-UO}_2$	2900 K	Ideal Binary Solution

6.3.4 Zr Oxidation Model

As a debris bed forms, the amount of Zr metal in the bed can be significant. If there is a supply of steam to the bed, the oxidation of the Zr can provide a significant heat source. In fact, if the cladding is rubblized, it will have a high surface to volume ratio and this will lead to a very large oxidation heat source.

The oxidation model examines the composition of the bed, the amount of steam and hydrogen flowing through the bed, and the temperature of the bed to determine the oxidation rate. The Zr particulate is assumed to be represented by an effective

spherical diameter. Once the rate of oxidation is determined, it is multiplied by the surface to volume ratio and heat of formation to determine the power source.

The kinetics of the oxidation are determined by using the Cathcart model for temperatures up to 1800 K and the Urbanic model for higher temperatures [6.8]. These kinetic models simply determine the rate of oxide production given an unlimited steam supply. The oxidation model uses this rate as an upper limit. If the steam supply or amount of Zr metal is insufficient to sustain this rate, then the rate is modified accordingly. The possibility of hydrogen blanketing is also considered as a limiting mechanism. Blockages in the bed are assumed to steam starve regions of the bed above their location.

6.4 Future Work

In a review [6.9,6.10] of the DEBRIS module in the MELPROG/MODO code, it was concluded that: (1) a two-dimensional debris meltdown model is needed to analyze accidents, such as Three-Mile Island, during which radial crust failure is possible; (2) a liquid momentum equation must be solved to accurately predict melt relocation; and (3) detailed phase diagrams are needed. A mechanistic, two-dimensional model was subsequently developed to analyze $\text{UO}_2\text{-ZrO}_2$ beds [6.11,6.12]. Recently, metals, such as Zr and Fe, were added to the analysis [6.13-6.15]. The new model is being implemented in the code. A detailed description of the governing equations follows.

Mass conservation equations are solved for each species. A two-dimensional (r-z) momentum equation accounts for melt relocation due to both gravity and capillary forces; viscous drag is included by modifying Darcy's Law to account for undersaturated flow, and the wetting behavior of molten stainless steel in contact with UO_2 is modeled using the formulation suggested by Scheidegger [6.16]. As solid melts in the center of the bed, the porosity increases; collapse of the bed is incorporated using a critical minimum solid volume fraction $\alpha_{s,\min}$. The energy equation considers conduction and radiation heat transfer in the bed, and phase diagrams are used to model Fe-Zr and U-Zr-O interactions.

6.4.1 Conservation of Mass

Mass conservation equations are solved for each species j : $j=1,2,3,4$ and 5 corresponding to UO_2 , ZrO_2 , Zr, stainless steel

and Ag-In-Cd. Balancing the mass stored in a differential control volume, convection by the liquid and the solid (as it collapses) and production (or depletion) gives

$$\begin{aligned} \frac{\partial}{\partial t} [\alpha_l Y_{lj} \rho_{lj}] + \bar{V} \cdot [Y_{lj} \rho_{lj} \bar{u}] + \frac{\partial}{\partial z} [\alpha_s Y_{sj} \rho_{sj} U] \\ = - \frac{\partial}{\partial t} [\alpha_s Y_{sj} \rho_{sj}] + R_j, \end{aligned} \quad (6.13)$$

where the subscripts l and s refer to liquid and solid, respectively, $\bar{u}=(u,v)$ is the liquid superficial velocity, U is the velocity at which the solid collapses, α is a volume fraction, Y_{ij} is the volume fraction of phase i that is occupied by species j, and R_j is the rate of production of species j by oxidation.

Note that species diffusion is neglected in Eq.(6.13). Typical diffusivities for liquids near their melting points are on the order of 10^{-9} m²/s [6.17]. For time scales on the order of 10^8 s, the distance characteristic of diffusion, $[Dt]^{1/2}$, is approximately 10^{-8} m. That is, liquid phase diffusion is only important over length scales comparable to the average particle diameter.

6.4.2 Liquid Motion

Balancing viscous drag (which is assumed to vary linearly with velocity), gravity and changes in pressure gives:

$$\rho_l \frac{\partial \bar{u}}{\partial t} + \frac{\mu}{\kappa_l} \bar{u} = - \bar{V} P_1 - \bar{g} \rho_l, \quad (6.14)$$

where μ is the dynamic viscosity of the liquid, g is the gravitational acceleration and κ_l is the relative permeability. Capillary forces enter Eq.(6.14) through the term involving the liquid pressure, P_1 . The capillary pressure is defined as the difference between P_1 and the gas pressure, P_g (that is, $P_c = P_1 - P_g$) [6.16]. Taking the gas flow to be isobaric gives $\nabla P_1 = -\nabla P_c$. From Eq.(6.14) it is therefore evident that capillary forces move liquid into regions of high P_c .

In order to solve Eq.(6.14) the relative permeability κ_l and the capillary pressure P_c must be specified. In fully saturated flow, κ_l equals the permeability κ , while in undersaturated flow, only a fraction of the solid is in contact with liquid and κ_l is proportional to κ ; the proportionality constant is a function of

the saturation S , which is defined as the fraction of nonsolid volume occupied by liquid (i.e. $S=a_l/(a_l+a_g)$). Reed [6.18] gives for κ_1 :

$$\kappa_1 = \begin{cases} \kappa S_e^3 & , \text{ for } S > S_r \\ 0 & , \text{ for } S \leq S_r \end{cases} \quad (6.15)$$

where $S_e = (S - S_r)/(1 - S_r)$ and the residual saturation S_r is defined as the threshold value of saturation below which bulk liquid motion ceases. For $S \leq S_r$, $\kappa_1 = 0$ and Eq.(6.14) requires that $\bar{u} = 0$. When the saturation S is less than the residual saturation S_r , the liquid consists of unconnected pendular rings. Liquid starts to flow when S is increased to the point that these rings touch and coalesce [6.19]. When the liquid wets the solid ($0^\circ < \theta < 90^\circ$),

$$S_r = \frac{1}{86.3} \left[\frac{\gamma \cos \theta}{\kappa \rho_l g} \right]^{0.263} \quad (6.16)$$

where γ is the surface tension and θ is the liquid-solid contact angle [6.7]. In a nonwetting system ($90^\circ < \theta < 180^\circ$), $S_r = 0.0$. The dependence of S_r on the porous matrix is contained in the permeability κ . Increasing the particle diameter or the porosity raises κ and leads to lower values of S_r . For packed beds consisting of small, tightly packed particles, the saturation must be increased to a high value before bulk liquid motion is observed. Bird [6.20] derives the following relation for κ by modeling the porous solid as a bundle of capillary tubes:

$$\kappa = \frac{d_p^2 \epsilon^3}{150 (1 - \epsilon)^2} \quad (6.17)$$

where the factor of 150 is determined empirically.

Leverett [6.19] derived the following equation for the capillary pressure P_c using dimensional analysis: $P_c = J \gamma \cos \theta (1 - a_s)^{1/2} / \kappa^{1/2}$, where J is a function of S_e only. Hoffman and Barleon [6.21] give an empirically determined relation for J : $J = a (S_e + b)^{-c}$, where $a = 0.38$, $b = 0.014$ and $c = 0.27$. Therefore,

$$P_c = a (S_e + b)^{-c} \gamma \cos \theta \frac{\sqrt{150} a_s}{d_p (1 - a_s)} \quad (6.18)$$

Because P_c decreases as the particle diameter d_p increases, capillary forces are small in beds with large particles. For $0^\circ < \theta < 90^\circ$, increasing the solid fraction α_s or decreasing the saturation S_e increases P_c ; capillary forces therefore move liquid into regions of high α_s (low porosity) and low saturation. For $90^\circ < \theta < 180^\circ$, decreasing α_s or increasing S_e increases P_c ; capillary forces therefore move liquid into regions of low α_s (high porosity) and high saturation. That is, melt tends to agglomerate in a nonwetting system. This effect has been observed in stainless steel- UO_2 beds in both the Molten Pool (MP) and Dry Capsule (DC) experiments [6.22,6.23].

6.4.3 Solid Motion

In order to determine the solid velocity U , the manner in which a high porosity bed collapses must be specified. It is assumed here that solid begins collapsing downward when the solid fraction falls below a critical value $\alpha_{s,min}$ and that $\alpha_s = \alpha_{s,min}$ in regions into which solid is collapsing.

The particle density n is determined from

$$\frac{\partial n}{\partial t} + \frac{\partial}{\partial z} nU = \Gamma \quad , \quad (6.19)$$

where Γ is the rate of destruction by phase changes. After solving Eq.(6.19) for n , the particle diameter is determined from $d_p = (6\alpha_s/n\pi)^{1/3}$.

6.4.4 Conservation of Energy

Balancing the energy stored in the solid and the liquid, convection by the liquid and the solid (as it collapses), diffusion and internal heat generation gives:

$$\begin{aligned} \frac{\partial}{\partial t} \sum_j [\alpha_s Y_{sj} \rho_{sj} h_{sj} + \alpha_l Y_{lj} \rho_{lj} h_{lj}] + \bar{V} \cdot [\bar{u} \sum_j Y_{lj} \rho_{lj} h_{lj}] \\ + \frac{\partial}{\partial z} [\alpha_s U \sum_j Y_{sj} \rho_{sj} h_{sj}] = \bar{V} \cdot k_{eff} \bar{\nabla} T + [\alpha_s \rho_{s1} Y_{s1} + \alpha_l \rho_{l1} Y_{l1}] Q \\ + q_c - a_v H (T - T_g) \quad , \end{aligned} \quad (6.20)$$

where h_{ij} is the enthalpy of species j in phase i , k_{eff} is an effective thermal conductivity which accounts for both conductive and radiative heat transfer in the porous solid, Q is the decay heat expressed as energy release per mass of UO_2 , q_c is the rate

at which energy is released by combustion, H is a heat transfer coefficient between the debris and vapor, T_g is the vapor temperature and a_v is the surface area of the debris per unit volume. It is assumed that the melt and the solid debris are in local thermal equilibrium (this is a reasonable assumption because melt velocities in the bed are low and the solid surface area per unit volume is large).

Radiation heat transfer in the packed bed is incorporated using a modified gas conductivity, $k_g^* = k_g + k_{rad}$, where $k_{rad} = 4\epsilon\sigma_d T_p^3$, ϵ is the emissivity of the solid and σ is the Stefan-Boltzmann constant [6.1]. Empirical correlations for k_{eff} in solid-gas systems are available in the literature [6.1]. However, in the current problem, three phases (solid, liquid and gas) are present. In calculating k_{eff} , the solid and liquid are treated as a single component with a volume-averaged thermal conductivity k_σ :

$$k_\sigma = \frac{1}{\alpha_s + \alpha_l} \left[\alpha_s \sum_j Y_{sj} k_{sj} + \alpha_l \sum_j Y_{lj} k_{lj} \right] \quad (6.21)$$

For a single phase i , k_σ depends only on Y_{ij} and k_{ij} . The following correlation gives k_{eff} [6.1]:

$$k_{eff} = \psi k_g^* + \frac{1-\psi}{k_\sigma \omega + k_g^* (1-\omega)} k_\sigma k_g^* \quad (6.22)$$

where $\omega = 0.3\alpha_g^{1.6} (k_g/k_g^*)^{-0.644}$, $\psi = (\alpha_g - \omega)/(1-\omega)$ and α_g is the volume fraction occupied by gas. In the limit as the gas volume fraction goes to zero, both ω and ψ approach zero and consequently, $k_{eff} = k_\sigma$. On the other hand, in the limit as the gas volume fraction goes to one, ψ approaches one and $k_{eff} = k_g^*$.

6.4.5 Phase Diagrams

At the present time, reaction rate information is not available in the literature for the materials and temperatures of interest. Consequently, equilibrium phase diagrams are used in this study to model Fe-Zr and U-Zr-O interactions. Shown in Fig. 6.1 is the Zr-Fe phase diagram given by Hansen [6.24]. Note that solid melts at a very low temperature for Zr atomic fractions greater than 0.33; when the temperature reaches 1200 K, a eutectic reaction occurs forming a Zr-Fe liquid solution having a Zr fraction of 0.76. For Zr atomic fractions between 0.33 and

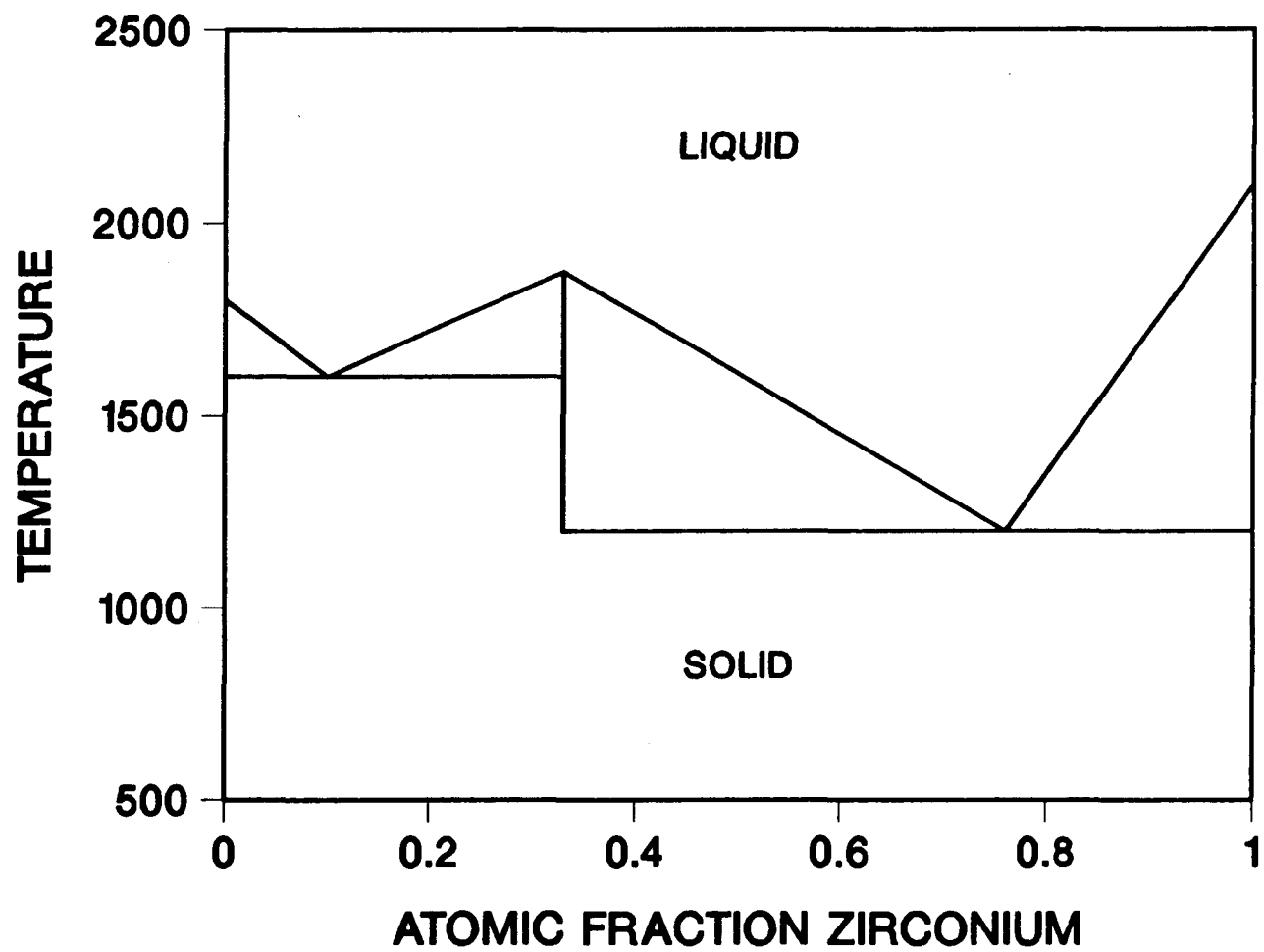


Fig. 6.1 Zr-Fe phase diagram given by Hansen [6.4]

0.76, this eutectic reaction decreases the Zr atomic fraction in the solid phase to 0.33. That is, a solid ZrFe_2 compound is created. For Zr atomic fractions between 0.76 and 1.0, the reaction increases the Zr fraction in the solid to 1.0 (all of the Fe liquefies). Because of the low solidification temperature (1200 K) of the Zr-Fe solution, only a small fraction will freeze near the bottom of the bed; most of the melt flows out of the bed very early in the meltdown sequence. This phenomenon is discussed further in Ref. [6.13].

When the temperature surpasses 2100 K, molten Zr dissolves both UO_2 and ZrO_2 [6.25,6.26]. In this study these dissolution processes are taken to be symmetric with respect to UO_2 and ZrO_2 compositions. For a given local (solid and liquid) composition, it is assumed that the melt composition falls along the line shown in Fig. 6.2. The solubilities of UO_2 and ZrO_2 in molten Zr are computed from the U-Zr-O liquidus temperature.

Although Zr- UO_2 , Zr- ZrO_2 and ZrO_2 - UO_2 pseudo binary phase diagrams are available in the literature [6.25,6.26], a ternary U-Zr-O phase diagram is not available. Ternary liquidus temperatures are determined in this study following the interpolation method suggested in Ref. [6.27]:

$$T_L = \frac{\sum_{\substack{i,j \\ i \neq j}} f_i f_j T_{Lij}}{\sum_{\substack{i,j \\ i \neq j}} f_i f_j}, \quad (6.23)$$

where the f 's are atomic fractions and T_{Lij} is the liquidus temperature for the i - j binary phase diagram. Equation (6.23) correctly reduces to $T_L = T_{Lij}$ when only two components are present.

6.4.6 Closure

In order to complete the preceding set of equations, oxidation rates, properties and initial conditions must be specified. Zirconium oxidation is modeled by the reaction $\text{Zr} + 2\text{H}_2\text{O} \rightarrow \text{ZrO}_2 + 2\text{H}_2$ and the kinetics data given by Prater and Courtright [6.28] are utilized. Properties are given by the MELPRO properties routine that is discussed in Section 10.

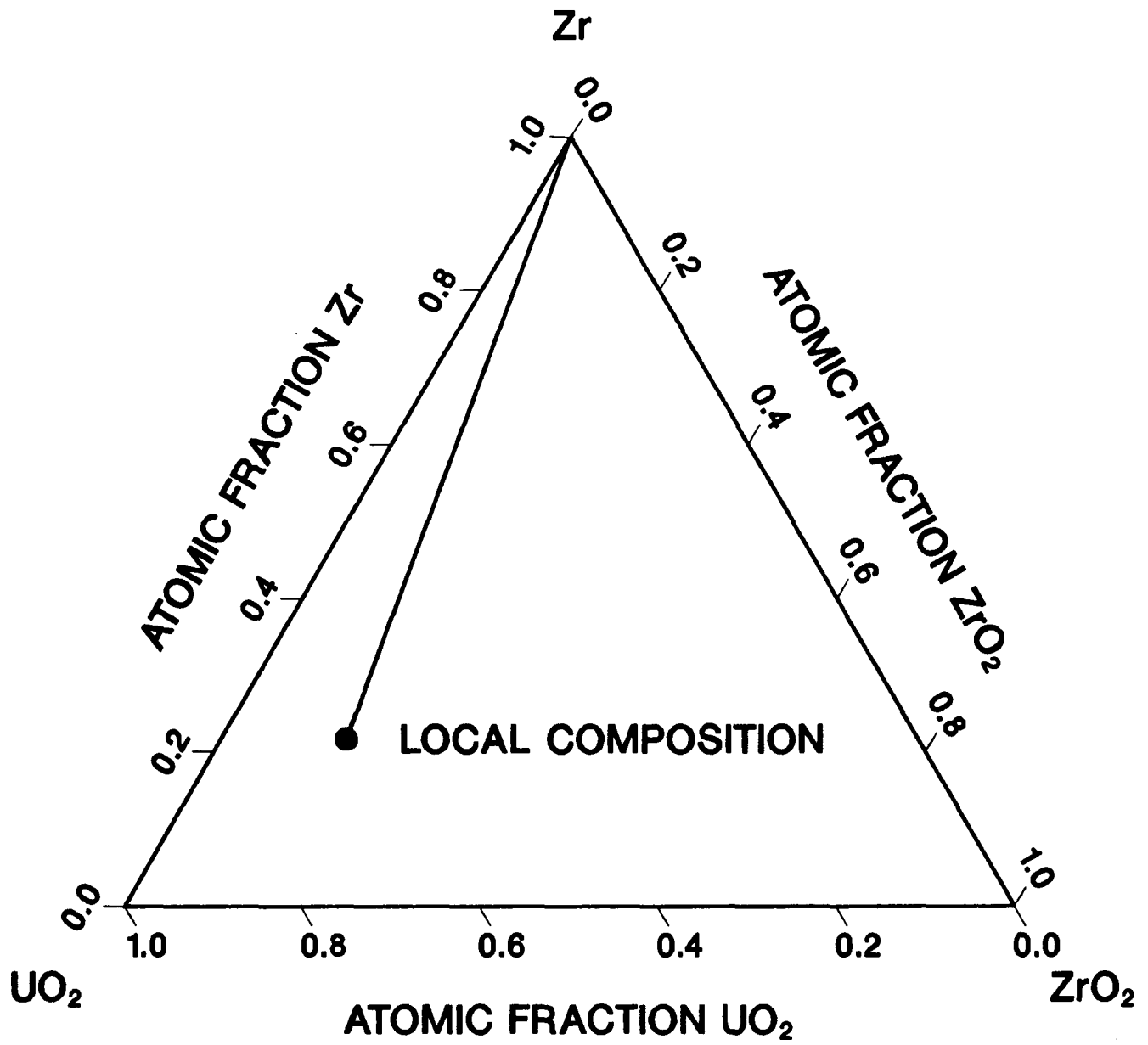


Fig. 6.2 Possible melt compositions for a given local (solid and liquid) compositions

6.5 References

- 6.1 J.E. Kelly, J.T. Hitchcock, and M.L. Schwarz, "Heat Transfer Characteristics of Dry Porous Particulate Beds with Internal Heat Generation," Proceedings of the ASME/JSME Thermal Engineering Joint Conference, Vol. 4, p. 83 (1983).
- 6.2 J.A. Moore, A Mechanistic Debris Bed Packing Model, LA-9619-MS, Los Alamos National Laboratory, Los Alamos, NM (1983).
- 6.3 S. Imura and E. Takegoshi, "Effect of Gas Pressure on the Effective Thermal Conductivity of Packed Beds," Heat Transfer Japanese Research, Vol. 3, No. 4, p. 13 (1974).
- 6.4 D. Vortmeyer, "Radiation in Packed Solids," 6th Int. Heat Trans. Conf., Toronto, Canada (1978).
- 6.5 G.P. Willhite, D. Kunii, and J.M. Smith, "Heat Transfer in Beds of Fine Particles (Heat Transfer Perpendicular to Flow)," AIChE Journal, Vol. 8, No. 3, p. 340 (1962).
- 6.6 A.V. Luikov, A.G. Shashkov, L.L. Vasiliev, and Yu. E. Fraiman, "Thermal Conductivity of Porous Systems," Int. J. Heat Mass Transfer, Vol. 11, p. 117 (1968).
- 6.7 Brown and Associates, Unit Operations, J. Wiley and Sons, New York, NY, p. 223 (1950).
- 6.8 P.J. Maudlin, et al, Light-Water Reactor Degraded-Core Cooling Program Technical Note A Damage Assessment of TMI-2, LA-UR-83-737, Los Alamos National Laboratory, Los Alamos, NM (1983).
- 6.9 S.S. Dosanjh, On the Melting and Refreezing of Porous Media, NUREG/CR-5028, SAND87-1976, Sandia National Laboratories, Albuquerque, NM (October 1987).
- 6.10 S.S. Dosanjh, "On the Melting of Reactor Core Particle Beds," 25th National Heat Transfer Conference, Houston, TX (July 1988).
- 6.11 S.S. Dosanjh, Melt Progression in Severely Damaged Reactor Cores, NUREG/CR-5029, SAND87-2384, Sandia National Laboratories, Albuquerque, NM (December 1987).

- 6.12 S.S. Dosanjh, "Melt Progression in Core Debris Beds," ANS 1988 Annual Meeting, San Diego, CA (1988).
- 6.13 S.S. Dosanjh, Relocation of Metallic Constituents in Core Debris Beds, NUREG/CR-5109, SAND88-0535, Sandia National Laboratories, Albuquerque, NM (1988).
- 6.14 S.S. Dosanjh and R.O. Gauntt, "MELPROG Debris Meltdown Model and Validation Experiments," 25th National Heat Transfer Conference, Houston, TX (July 1988).
- 6.15 S.S. Dosanjh, "A Detailed Mechanistic Analysis of Core Debris Melt Progression Behavior in LWR's - The MELPROG DEBRIS Module," ENS/ANS International Conference on Thermal Reactor Safety, Avignon, France (October 1988).
- 6.16 A.E. Scheidegger, The Physics of Flow Through Porous Media, Third Edition, pp. 266-290, University of Toronto Press, Toronto, Canada and Buffalo, NY (1974).
- 6.17 C.A. Wert and R.M. Thomson, Physics of Solids, Second Edition, McGraw-Hill Book Co., New York, NY, pp. 71-72 (1970).
- 6.18 A.W. Reed, et al., DCC-1/DCC-2 Degraded Core Coolability Analysis, NUREG/CR-4390, SAND85-1967, Sandia National Laboratories, Albuquerque, NM (October 1985).
- 6.19 M.C. Leverett, "Capillary Behavior in Porous Solids," Pet. Trans. AIME 142, pp.152-169 (1941).
- 6.20 R.B. Bird, W.E. Stewart, and E.N. Lightfoot, Transport Phenomena, John Wiley and Sons, Inc., New York, NY, p. 119 (1960).
- 6.21 G. Hofmann and L. Barleon, "Reduced Coolability of Particle Beds as a Result of Capillary Effects at Horizontal Phase Boundaries," Proceedings of the International ANS/ENS Topical Meeting on Thermal Reactor Safety, San Diego, CA (February 2-6, 1986).
- 6.22 J.T. Hitchcock and J.E. Kelly, "Post-Test Examinations of the In-Pile Molten Pool Experiments," Trans. Am. Nucl. Soc. 43, p. 515 (1982).

- 6.23 C.P. Fryer and J.T. Hitchcock, Postirradiation Examination of the DC Melt Dynamics Experiments, NUREG/CR-4625, SAND86-1102, Sandia National Laboratories, Albuquerque, NM (July 1988).
- 6.24 M. Hansen, Constitution of Binary Alloys, Second Edition, McGraw-Hill Book Company, New York, NY, pp. 741-743 (1958).
- 6.25 D.L. Hagrman, G.A. Reymann and R.E. Mason, MATPRO-Version 11 (Revision 2): A Handbook of Material Properties for Use in the Analysis of Light Water Reactor Fuel Rod Behavior, NUREG/CR-0497, Idaho National Engineering Laboratory, Idaho Falls, ID (1981).
- 6.26 D.L. Hagrman, Material Property Models for Severe Core Damage Analysis, EGG-CDD-5801, Idaho National Laboratory, Idaho Falls, ID (1982).
- 6.27 Reactor Safety Research Semiannual Report January-June 1987 Volume 37, pp. 275-280, NUREG/CR-5039 (1 of 2), SAND87-2411 (1 of 2), Sandia National Laboratories, Albuquerque, NM (January 1988).
- 6.28 J.T. Prater and E.L. Courtright, High-Temperature Oxidation of Zircaloy-4 in Steam and Steam-Hydrogen Environments, NUREG/CR-4476, PNL-5558, Pacific Northwest Laboratory, Richland, WA (1986).
- 6.29 Surry Power Station Final Safety Analysis Report (FSAR), Virginia Electric and Power Company (VEPCO) (Dec. 1969).

7.0 VICTORIA MODULE

As discussed in the introduction to this report, the release of radionuclides into the environment has been identified as the primary risk associated with the operation of nuclear reactors [7.1]. The VICTORIA subcode will calculate all aspects of fission product release and transport within the vessel and the Reactor Coolant System (RCS). Because of the importance of the VICTORIA module, a detailed description is provided in this report even though this module is not yet fully coupled to other modules in MELPROG. The principal current method of running VICTORIA is as a stand-alone model. However, a one-way coupling exists with the FLUIDS, CORE and STRUCTURES modules; these modules provide VICTORIA with information it needs. In the future, the VICTORIA module will calculate the decay heat for the other modules.

The initial inventory of fission products in the reactor fuel and the decay heat associated with them are provided at the start of a calculation by ORIGEN [7.2]. The major physical states for which VICTORIA is to calculate release are then rods, solid rubble, melting rubble, molten pools, rapidly fragmented and quenched liquids, as during a steam explosion, and surfaces on which fission products have condensed (revaporization). In addition, the reentrainment of previously deposited aerosols during high pressure blowdowns is an important process which must be treated by VICTORIA/MELPROG.

The ultimate driving force for the release of fission products is the inherent tendency of physical and chemical systems to move toward a state of thermochemical equilibrium. Thus, whenever the partial pressures of fission product chemical species in the coolant vapor overlying the fuel are lower than their equilibrium partial pressures, there exists a driving force for release. Conversely, whenever the partial pressures existing above any surface are higher than the equilibrium partial pressures above that surface, there is a driving force for condensation on the adjacent surface. The tendency for release is hindered by a number of resistances acting in series and parallel. First, release is hindered by the finite rate at which molecules are transported from within the fuel to its surface. Any overlying clad forms another barrier to release. Transport within the fuel-clad gap to a clad failure location is yet another resistance. Transport across a convective boundary layer is a further hindrance to release. Another potential resistance,

vaporization kinetics, is adjudged to seldom--if ever--be important.

Once released, the fission products are transported by FLUIDS. This transport can be as vapor or on aerosols (either self-nucleated from fission products or preexisting). VICTORIA also tracks (in concert with FLUIDS) vapor condensation on structures and aerosol deposition. Once condensed, vapors may reheat the surface or aerosol they are attached to. This in turn can lead to revaporization. Finally, re-entrainment of deposited aerosols is treated.

Release from all physical states is treated on a similar thermochemical and transport basis. However, the actual physical transport limitations vary greatly from one state to another. For intact rods, transport within the fuel can be limiting because of relatively low temperatures. As well, the equilibrium vapor pressures themselves may be low enough to limit release; a largely intact clad is an obvious major barrier to release. However, mass transport in the vapor phase over intact rods is almost never limiting. By contrast, for rubble beds, the small characteristic particle size (high surface to volume ratio) and high temperatures means that neither transport in the fuel nor low equilibrium vapor pressure is likely to constitute a rate limitation for many species. However, the very low coolant gas flow velocities in rubble beds means that mass transport limitations in the vapor phase are of crucial import. Finally, for a large molten pool, the most important limitation to release arises from the large characteristic transport distances within the pool (small surface-to-volume ratio).

The structure and framework of the VICTORIA module is discussed in Section 7.1. Sections 7.2 and 7.3 describe species behavior within the fuel and the bulk gas, respectively. Subsequent sections are concerned with equilibrium chemistry and radioactive decay heating. Recently, the CHARM code [7.3] was developed to analyze aerosol behavior and incorporated into the developmental version of VICTORIA; this model is briefly discussed in Section 7.6.

7.1 Methodology and Framework of the Victoria Module

The purpose of this section is to give the reader a general overview of the calculational method of the VICTORIA code. Basically the code attempts to model the behavior of chemical

species (with modeling emphasis on radioactive fission-product species) in the primary system of a Light Water Reactor (LWR) during a severe accident transient. This accident transient is assumed to subject the core to high temperatures, high pressures, large amounts of metal oxidation, and substantial internal-structure geometry change. To effect the modeling of the behavior of chemical species in this severe accident environment, systems of conservation equations are solved that describe the transport, phase change, and chemical exchange experienced by VICTORIA's species set.

7.1.1 Solution Domain

The spatial domain that the VICTORIA code addresses includes the interior parts of a reactor vessel. In particular, the fuel rods, structural surfaces, and the volume normally occupied by the reactor coolant represent VICTORIA's domain. Since structural geometry varies in a severe accident transient, this space domain also varies: fuel rods degrade into distorted shapes and possibly into liquids or debris, structures melt and relocate, and the coolant volume obviously can increase in some regions and decrease in others. In this report, fuel pellets or debris that have intact or degraded cladding, or no cladding at all, will be referred to as the "fuel geometry." There exist other ongoing efforts to extend the space domain to include the entire primary system (i.e., the MELPROG/VICTORIA coupling efforts), but that work will not be discussed here.

In order to numerically solve the species transport equations in the FLUIDS module, the vessel interior is discretized into a two-dimensional cylindrical (r,z) Eulerian mesh, illustrated in Fig. 7.1. This mesh scheme normally has its origin at the bottom-center of the vessel (although it can include the reactor cavity as well), the z axis represents the vessel center-line and the scheme can contain an arbitrary number of radial cells (rings) and axial cells (levels of lengths Δr and Δz , respectively). The sample mesh in Fig. 7.1, which is much coarser than that normally used in actual calculations, has four radial rings and six axial levels, with the cross-hatched area representing the core region, an upper plenum above the core, a

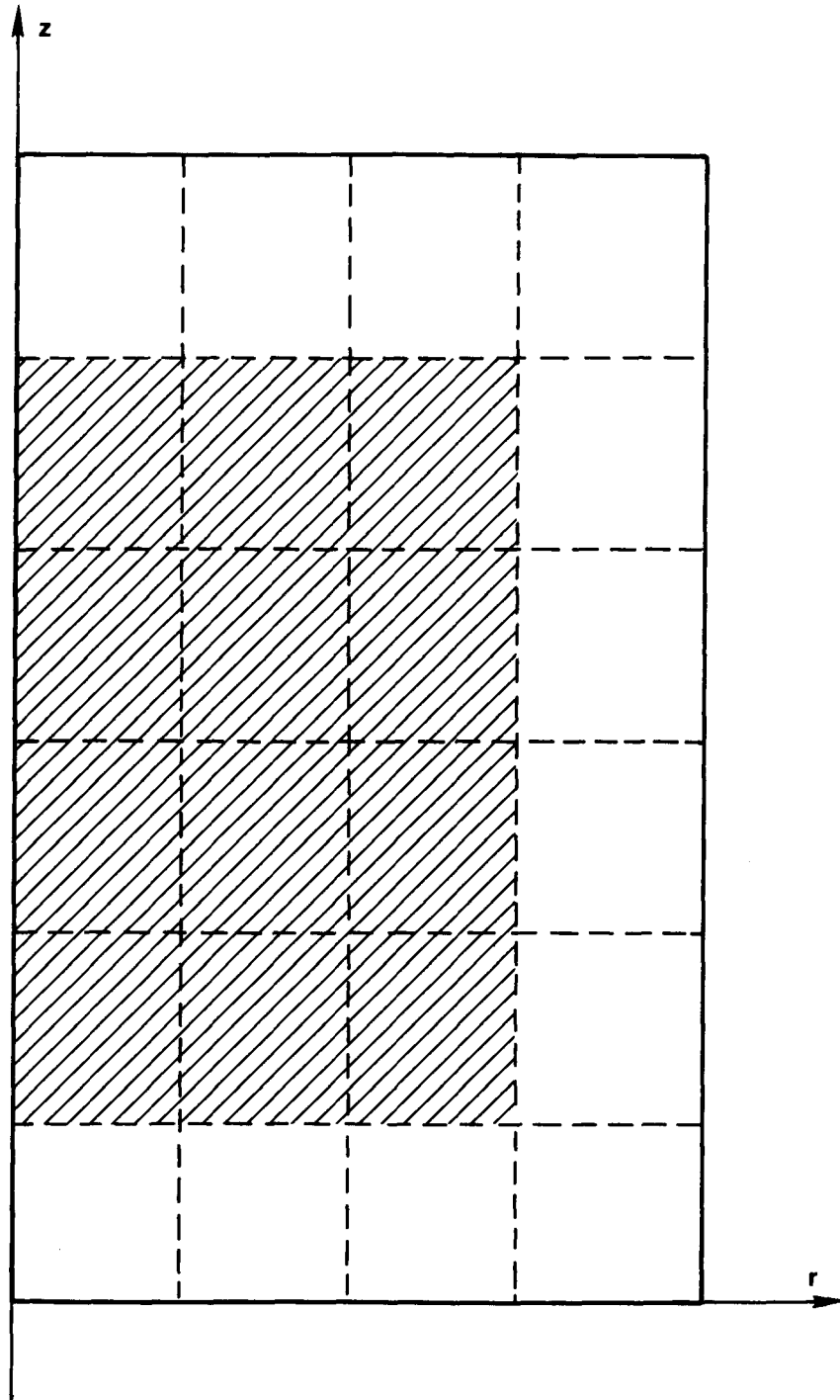


Fig. 7.1 Simple mesh cell diagram dividing the reactor vessel into six axial levels and four radial rings. The cross-hatched area designates the core region.

lower plenum below the core, and a vessel downcomer at the right periphery (note that this illustration is not to scale). Boundary conditions must be prescribed at all boundaries of the domain depicted in Fig. 7.1.

Within each of these 2D FLUIDS cells there are sub-domains that discretize the fuel geometry into two-dimensional cells. Within these cells the structure surfaces are treated as either a horizontal or a vertical surface area. The fuel geometry can be treated as a 2D cylinder, or a 1D sphere. The actual choice depends on the fuel geometry specified by MELPROG. For intact rods or rod segments the 2-D cylindrical geometry is the better model while for fuel debris the 1-D spherical geometry is superior. Also, within each FLUIDS cell there are so-called "chemistry regions" that represent volumes of space where the species are allowed to experience chemical change. At the present time there are a maximum of five chemistry regions per gross cell: the fuel grains, the fuel open porosity, the fuel/cladding gap which includes the inner clad surface, the bulk gas which includes the aerosol particles, and the structure surface. Logically if one of these regions does not exist within the FLUIDS cell, then the associated chemistry is not performed. Subsequent sections will illustrate these points more clearly.

The time domain in VICTORIA is discretized into a series of time-steps (Δt) whose size is controlled by the spatially coupled transport terms (convection and diffusion) in the conservation equations. Although some parts of the conservation equations are solved implicitly, the time step must be constrained by a material Courant condition [7.4] because of the presence of explicit source terms in the equations.

Another independent variable that can be associated with the solution domain is the discrete species index. The total number of species that can be treated by VICTORIA is arbitrarily large; the maximum number is practically determined by available computing resources. At the present time VICTORIA operates with 99 species.

7.1.2 Modeling Overview

As mentioned above the numerical approach in the code discretizes the solution time domain into a series of time-steps, and during each of these steps systems of equations are solved over the spatial mesh that describe the following series of perceived coupled physical phenomena:

1. species release from the fuel geometry controlled by chemistry and transport mechanisms;
2. species behavior in the reactor coolant volume in terms of gaseous transport, aerosol evolution and transport, and chemistry; and
3. species behavior on structural surfaces which considers aerosol deposition, mass transport, and chemistry.

This collection of modeled phenomena is illustrated in Fig. 7.2, which portrays schematically the most complicated situation for a computational cell: a fuel domain from which species are being released into a bulk gas environment that contains aerosols and structure surfaces. The dashed outer line in Fig. 7.2 designates the 2D cell boundary across which there exists gas and aerosol transport. The species diffusion shown at the ends of the fuel rod implies two-dimensional transport for the fuel geometry sub-domain. Practically speaking, many of the computational cells in the vessel domain will contain only a subset of the phenomena shown in Fig. 7.2 at any given time; for example the upper plenum cells will not contain fuel material and thus will not have to contend with the associated fuel geometry transport and chemistry mentioned in item 1 above. As an aside it is worth noting that if one were to disregard or appropriately model the species release from the fuel geometry (application specific to reactor safety), VICTORIA could be used in any application that contains a species continuity problem with chemistry and aerosol behavior.

All of the phenomena illustrated in Fig. 7.2 are spatially dependent, but only the transport terms are spatially coupled. For clarification purposes a distinction is made here between intra-cell (r, z dependent) and inter-cell (gradient dependent) phenomena. At the current time only diffusive and convective transport terms in VICTORIA's modeling are spatially coupled.

7.1.3 Calculational Flow

The calculational framework contained within the VICTORIA module consists of four main parts or sub-modules. These are:

1. equilibrium chemistry calculations for all species located within the fuel geometry, the bulk gas, the aerosol particles and on all structure surfaces;
2. a behavior and transport calculation of the species within the UO_2/Zr fuel geometry;

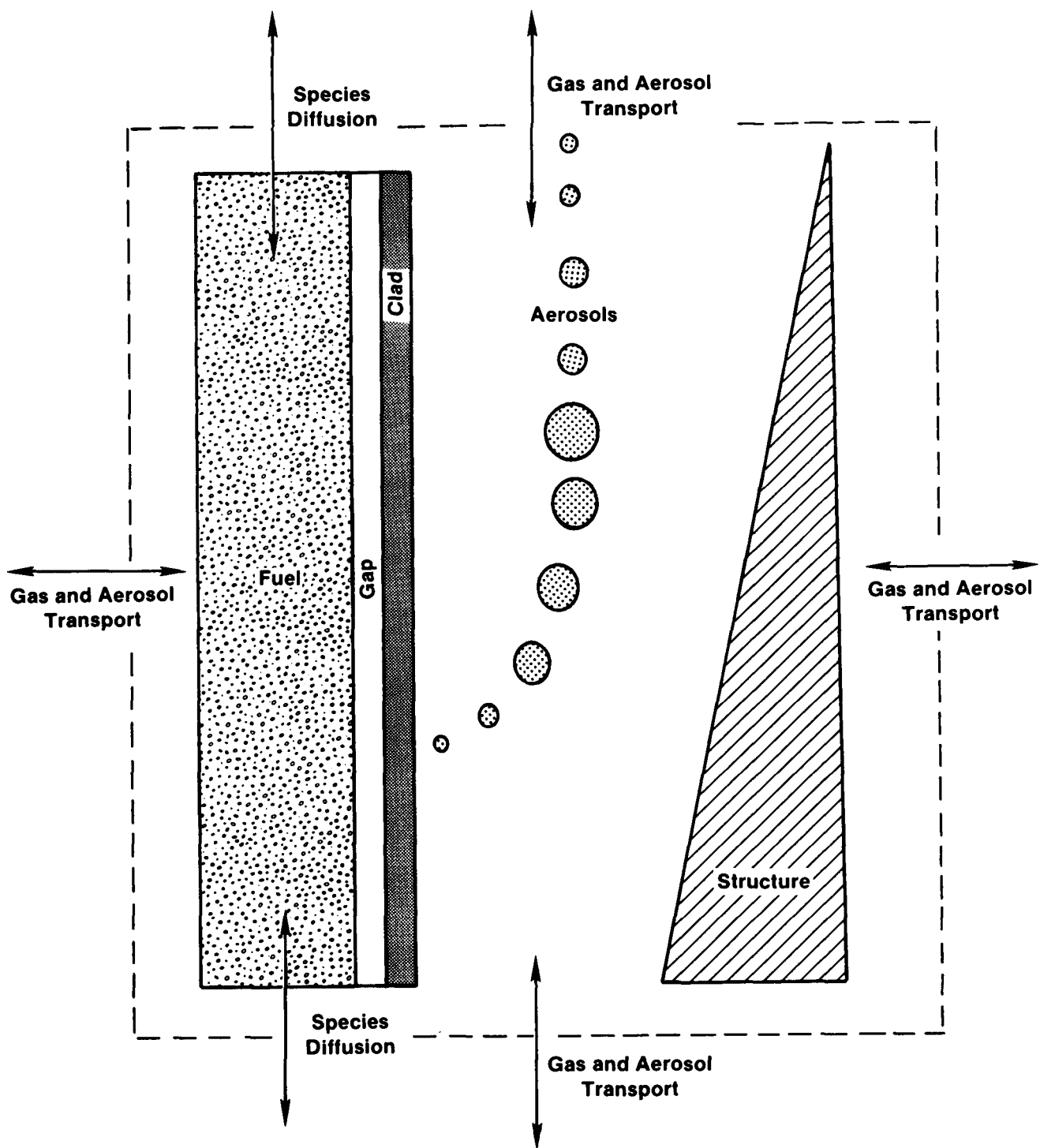


Fig. 7.2 Schematic of the relevant geometry and phenomena related to species behavior and transport which are modeled within each gross two-dimensional mesh cell.

3. a transport calculation of the species in the bulk gas medium; a species continuity calculation on all structure surfaces including the fuel geometry surface; and
4. an aerosol formation, size evolution, transport and deposition calculation in the bulk gas medium.

This framework is illustrated in Fig. 7.3. It is important to recognize that each of these calculational sub-modules (i.e., each of the boxes in Fig. 7.3) solves its own internal system of equations explicitly coupled to the other sub-modules. This means that communication between sub-modules is not instantaneous but is typically one time-step out-of-phase. A good example of the explicit coupling is the solution of the diffusion equation in the fuel geometry which uses bulk gas boundary conditions that are Δt old. The physical couplings between the sub-models in Fig. 7.3 are explained in more detail in subsequent sections. Also indicated in Fig. 7.3 are subroutine names that represent the drivers for the calculations associated with each sub-module.

The Initial Species Inventory box in Fig. 7.3 represents the acquisition of the initial species concentrations. This information is currently obtained by post-processing the output from an ORIGEN calculation [7.2], although the results produced by any isotopic depletion code would supply the required initial condition.

Figure 7.4 illustrates the calculational flow (i.e., the sequence of subroutine calls) currently used by VICTORIA. This sequence of calls is looped over during each time-step in the main driver routine VIKI as indicated by the "A" loop in the figure. The names of the main physics sub-modules (from Fig. 7.3) that are called from inside this loop are also included in parentheses.

As mentioned above, it is important that VICTORIA's time-step be constrained to a material Courant condition that is defined here to include both convective and diffusive transport terms ($D_{\max}/\Delta r$ can be thought of as a diffusion velocity). In simple terms this condition has the form:

$$\Delta t_{\text{bulk}} = \text{Min} \left[\frac{\Delta r}{u}, \frac{\Delta r^2}{D_{\max}}, \frac{\Delta z}{v}, \frac{\Delta z^2}{D_{\max}} \right], \quad (7.1)$$

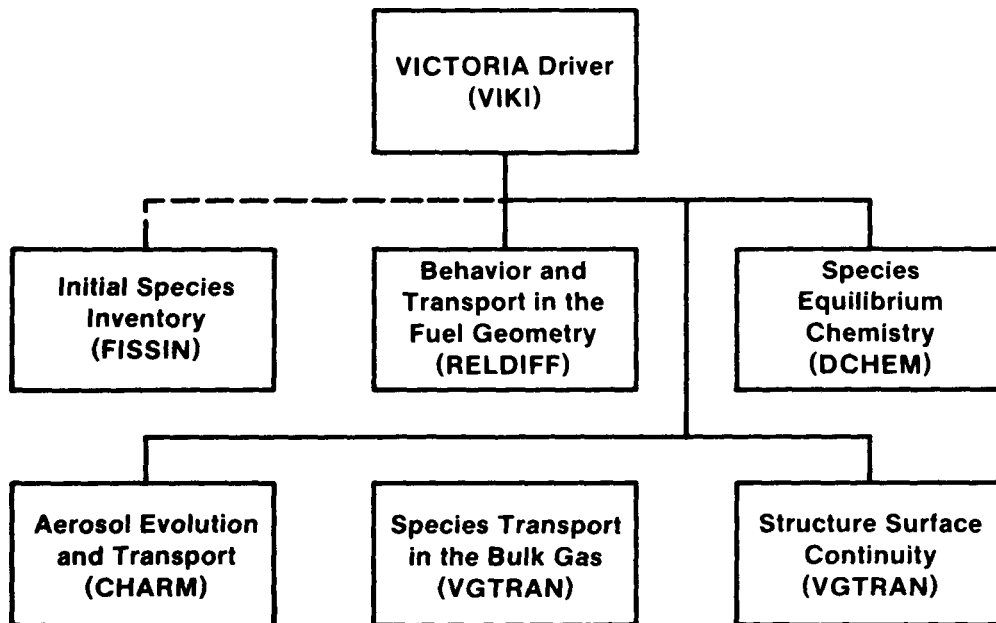


Fig. 7.3 The program structure of the VICTORIA module. Each box depicts an area of modeling that solves its own internal system of equations coupled explicitly to the other modeling areas (other boxes). The dotted line indicates a single subroutine call.

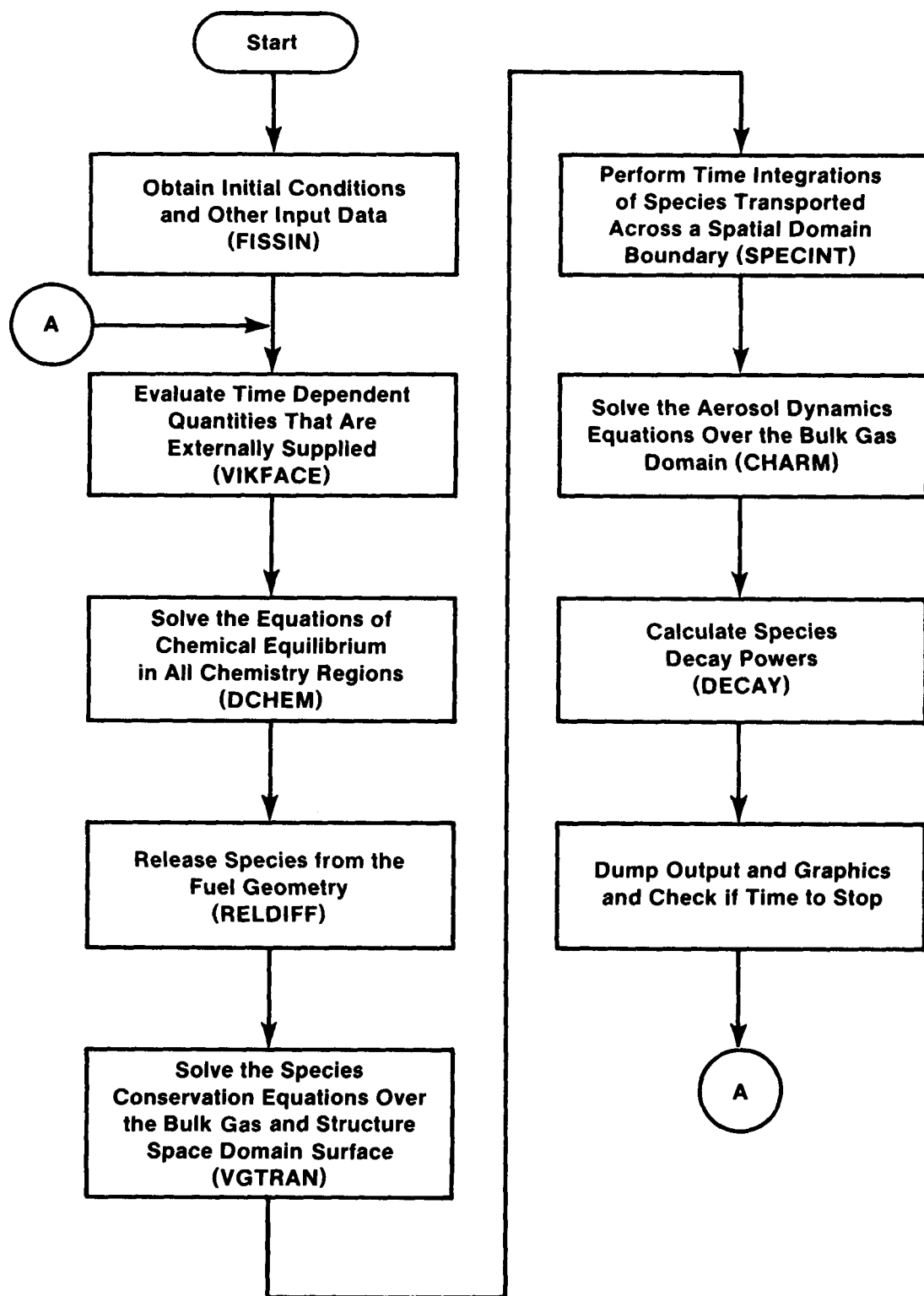


Fig. 7.4 Flowchart of the logic flow in subroutine VIKI.
Main subroutine names are indicated in parentheses.

$$\Delta t_{\text{fuel}} = \text{Min} \left(\frac{\Delta r^2}{D_{\text{max}}}, \frac{\Delta z^2}{D_{\text{max}}} \right), \quad (7.2)$$

$$\Delta t = \text{Min} (\Delta t_{\text{bulk}}, \Delta t_{\text{fuel}}), \quad (7.3)$$

where Δt_{bulk} is a bulk-gas cell time-step limit, Δt_{fuel} is a fuel-geometry cell time-step limit, Δr and Δz are the radial and axial cell dimensions, u and v are the radial and axial gas velocity, and D_{max} is the largest of the species' diffusion coefficients in the geometry domain of interest. Note that Eqs. (7.1) and (7.2) contain different numerical and property information. The Min function appearing in Eqs. (7.1) and (7.2) implies that the arguments are evaluated for all cells in the spatial domain and that the smallest value is then selected.

7.2 Species Behavior and Transport within the Fuel Geometry

The fuel geometry for the VICTORIA code is taken to include the fuel pellets and cladding. The calculation treats two separate physical regions. The first of these is the interior of the fuel grains where equilibrium chemistry and intragranular transport are treated. The second region can best be described as extra-granular and encompasses the open porosity, gap, and cladding along with their mutual and individual boundaries.

Within the grains (which are treated as uniform spheres) a simple Fick's Law diffusion equation is used to determine a source for the main transport equation that is solved within the extragranular region. Other sources and sinks for the governing equation are generated by the equilibrium chemistry solver and by clad oxidation and relocation models in the sub-regions of the extra-granular region. The only transport mechanism considered within the extra-granular region is diffusion. In any axial level that contains the fuel, pellets are considered to be homogeneous. Both gas phase and surface diffusion are treated depending on the physical state of the diffusing species.

The extra-granular region is divided into a number of sub-regions. These sub-regions are the open porosity, the gap, the clad, and the clad outer surface. A typical extra-granular nodding diagram is shown in Fig. 7.5. The axial dimension for all cells and the axial nodding (i.e., the number of cells in the axial direction) is specified in the input. The open porosity, gap, and clad have a radial dimension, as well, that is set by

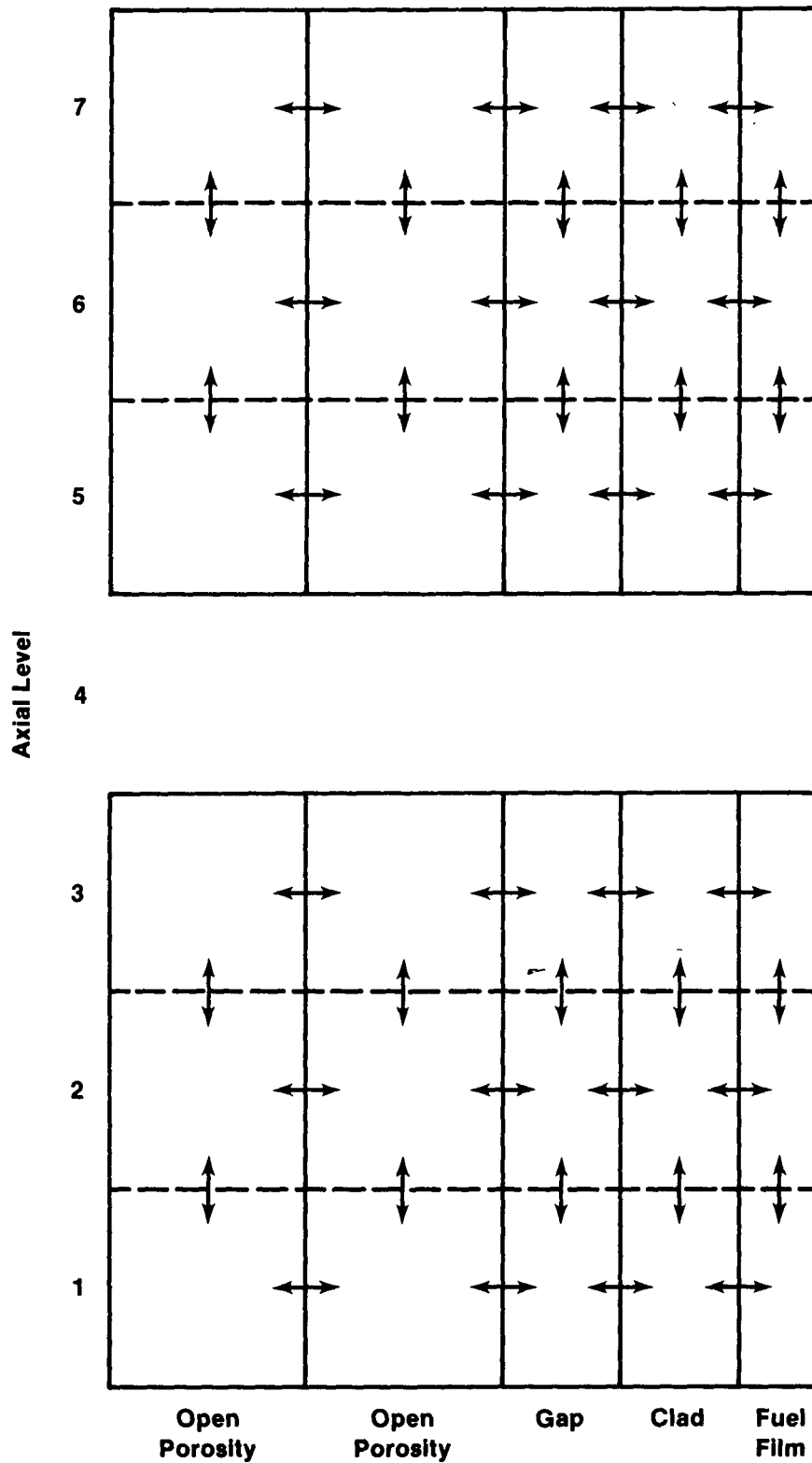


Fig. 7.5 Sample of the type of mesh used in the VICTORIA diffusion calculation.

the user in the input or can be obtained from the CORE module. In addition to the physical dimensions the user is required to set the number of mesh cells in the radial direction to be occupied by the open porosity and the cladding. The values assigned to these variables are based on a trade-off between computational efficiency (i.e., computing time) and accuracy in the transport routines. A larger number of radial nodes gives greater accuracy but at the expense of computer time. The value set for the open porosity also determines the setting for the fuel grains. In the example shown in Fig. 7.5 the open porosity occupies two radial mesh cells and the cladding occupies one. The gap is always set to a mesh width of one. The clad-coolant interface cell also has a mesh width of one and a radial dimension that is hard-wired in the code.

In a two-dimensional solution to the governing equation over this noding pattern each mesh cell communicates with the cells immediately above and below it along with those adjacent to it on the left and right. The exception to this occurs when there is no fuel element in an axial level. In this case there is no vertical communication between this level and its neighbors. In fact no fuel region transport is calculated in such an axial level.

7.2.1 Governing Equations

The transport processes within the extra-granular portion of the fuel are treated by the solution of one general transport equation. This equation is solved over the entire region for each chemical species. The solution is obtained independently for each species at each time step.

The general transport equation that is solved in this region is a Fick's Law diffusion equation. The driving force is taken to be the concentration gradient only and there is a source/sink term for intragranular fission product transport leading to release from the grains to the extragranular region. Chemistry leads to changes in the species concentrations, which are the driving force for diffusion, and thereby influence the diffusive transport. The equation that is solved is

$$\frac{\partial N_i}{\partial t} = -\nabla \cdot (-D_i \nabla N_i) + S_i^g \quad (7.4)$$

where N_i is the molar concentration of the i^{th} species, D_i is the species-dependent diffusion constant, and S_i^g is the source of species i escaping from the fuel grains. The values of the source terms and the form of the diffusion constant are dependent on the spatial region within the fuel as well as on the species phase and identity. The dependencies of the source term and the diffusion constant will be discussed in the sections that follow.

7.2.2 Intragranular Processes

The initiation of the release of fission products in a nuclear reactor accident takes place within the fuel grains. A variety of mechanisms have been postulated which lead to the movement of fission products from the inside of the grains to the surface of the grain and subsequent transport through the extragranular portion of the fuel. In the current version of VICTORIA we consider primarily two intragranular phenomena. The first is equilibrium chemistry which can severely affect the speciation of the fission products within the solid matrix and hence their effective mobility in this environment. The second phenomenon is the actual movement of fission products within the grains. In the current version of the code only solid state diffusion of atomic species is considered. Both of these topics will be discussed in greater detail in the following sections.

7.2.2.1 Equilibrium Chemistry in the Fuel Grains

Within the fuel grains the concentration of fission products is sparse but not negligible. It has been shown experimentally that the fission products do not all exist in their elemental form within the lattice [7.5]. Since the motion of the fission products within the ceramic lattice will be affected by their chemical form, some determination of this form has been undertaken in the code.

The simplest approach to chemical speciation is obviously one of neglect, that is, the fission products exist only in atomic form within the lattice. Since this has been shown not to be the case [7.5] one needs to use a more complex approach. The next simplest solution to the problem is the assumption of complete thermo-chemical equilibrium within a defined set of chemical species. A modified version of this approach is taken in the current version of the code.

At present the chemical equilibrium package in VICTORIA consists of a set of ninety-nine (99) chemical species made up from fourteen (14) elements. The identity of the species and elements is given in Section 7.4.3. The chemical equilibrium is calculated in an ideal solution consisting of the elements present in the fuel at the time of interest. For a mixture as complex as partially burned-up ceramic fuel, the ideal solution assumption is probably not the best possible choice. Studies of irradiated fuel [7.5] have shown that there are several solid phases present, each containing a number of different species. It is not clear that each or any of these phases satisfies the conditions of equilibrium for an ideal solution model. Another choice would be some sort of nonideal solution model based on data from studies of irradiated fuel as a function of oxygen to metal ratio in the fuel, burnup and power history. Such a model would require corrections to the free energy data as a function of the temperature and the concentrations of all the species present as well as those due to the physical parameters pertinent to the particular fuel specimen being modeled. Even if the data for such a calculation were available, it would be extremely difficult to carry out economically.

The small concentration of fission products within the lattice and the relatively low mobility of atoms within the lattice raises further the questions as to the validity of the full equilibrium approximation. A simple calculation, based on random walk statistics allows one to determine the collision frequency of atoms in the lattice. Using the solid-state diffusion coefficients of O_i and O_i and Takage [7.6] and typical concentrations and temperatures, it is estimated that collisions between fission product atoms take place on a 50 msec timescale. This calculation does not take account of the requirements for the formation of a molecule involving more than two fission product atoms. For such a molecule to form, it is necessary for an atom to find an existing chemical moiety which will not, in general, be thermodynamically favored. This precursor entity will therefore have a very small concentration, which reduces the probability for the necessary collision to occur to a level below that of a three body collision. This small probability for the requisite collision reduces the likelihood of formation of such molecules to the point where equilibrium is, at best, a tenuous assumption. This likelihood is even further reduced by the concept of orientation, that is, the atom not only has to find the existing molecule but the approach must be in a suitable

direction for new molecule formation. As a result of these considerations nonoxide species which consist of more than two fission product atoms are not allowed to form.

Species which consist of two fission product atoms (either the same or different) are not subject to the same steric or collisional probability factors as the species considered above. There is, however, a different steric constraint which must be taken into account. This constraint is a question of molecular size and the availability of a suitable lattice site for molecule formation. Non-oxides, in particular iodides or tellurides, consisting of two fission product atoms are, in general, much larger molecules than oxides. As a result, the ceramic lattice spacing is too small to allow formation of these molecules within the lattice. This size mismatch can be overcome by the existence of a suitable defect in the lattice or by the formation of a separate phase. The formation of a separate phase is, in our formulation of the problem, equivalent to migration of the fission product atoms from the grains into the open porosity region of the fuel. This deviation from equilibrium of fission product diatoms has been observed experimentally [7.7]. A modification has been made to the free energy of the diatomic iodides and tellurides in the fuel grains to account for this deviation.

The formation of oxides (e.g., simple binary oxides, uranates, zirconates, and molybdates) containing fission products is treated in an ideal solution, equilibrium manner. These species are observed in irradiated fuel [7.5] and there is no evidence that their concentration is not at thermodynamic equilibrium. It must be noted, however, that these species can exist in a separate phase consisting of one or more species. This phase separation is clearly an indication that a single ideal solution is not present. The existence of the oxides does demonstrate the need for a model to account for chemistry within the fuel grains.

The general treatment of the chemistry within the fuel grains follows a slightly different set of assumptions than that in other regions. The volume used is that of the fuel grains. The volume of the gas phase in this region is taken to be the same as that for the condensed phase rather than a different volume. The reasoning for this treatment has to do with the concept of phases within a solid solution. An isolated atom or molecule within the lattice is termed gaseous or condensed based solely on its free

energy and not on its actual physical phase. Under lattice conditions all atoms or molecules are essentially condensed since there is no separate region in which a bulk gas exists in equilibrium with the condensed material and occupies a different region of space. The release models used for motion within the lattice do not differentiate between condensed or gaseous atoms, and molecules are immobile regardless of their physical or thermodynamic phase.

This last statement concerning the immobility of molecules within the fuel grains delineates the primary affect of chemical interaction on the release process. What is occurring in the chemistry is the reduction of concentration of fission product atoms. Since the concentration gradient between the grain interior and grain surface is the driving force for the solid state diffusion process, the lessening of fission product atom concentration within the grains reduces the diffusion rate and hence meters the transport to the open porosity. A similar effect could be produced by using a smaller value of the diffusion coefficient and neglecting chemical interaction within the fuel.

Neither the modified ideal solution chemistry that is used in the code at present nor a complete neglect of fuel chemistry with concomitant modification of the solid state diffusion constants to account for the change in atomic concentration, is an accurate model of the actual physics that is occurring in the fuel leading to fission product release. It is likely, however, that either of these models, combined with an accurate time and temperature dependent model of fuel morphology, will reproduce observed releases to an acceptable accuracy. Furthermore, this solution to the problem does reflect the level of experimental knowledge of the actual process of fission product release from degraded fuel.

7.2.2.2 Solid-State Diffusion

Transport of fission products within the fuel grains is treated as taking place by solid state diffusion through the lattice. The driving force for this motion is the concentration gradient between the interior and the surface of the fuel grains. The resulting concentration at the surface of the fuel grain is used as a source term for the primary transport equation. The new concentration at the grain surface is determined from the equation

$$S_g = \frac{24D(N_g - N_s)}{\alpha d^2} \quad (7.5)$$

where N_g is the concentration within the grain, N_s is the concentration at the grain surface, d is the diameter of the grain, and α is the ratio of the volume of the open porosity to the fuel grain volume. The diffusion constant D is taken to be a simple function of the temperature for each diffusing species. The functional form used is

$$D = D_0 e^{(-\theta/T)} \quad (7.6)$$

where D_0 is in meters squared/second and θ is the activation energy for diffusion in degrees Kelvin. The values of D_0 and θ are specified in the input for each species. At present, the values used for these parameters are taken from Ref. [7.6]. It is intended that as the code matures a uniform set of these variables can be determined.

The species for which solid state diffusion is allowed are specified by the user in the input deck. It is recommended that only atomic species, either gaseous or condensed, be allowed to diffuse through the lattice. It is generally observed that molecular species diffuse through solids at rates that are much slower than atoms and would not be expected to contribute to release. The one possible exception to this would be the basic lattice constituent, UO_2 in this case, which can be assigned a diffusion coefficient to account for sublimation of the lattice. In this case, the activation energy should correspond to the heat of sublimation of UO_2 . For the high temperature regime typically operative in a severe accident a considerable amount of fuel could be sublimed. This is dependent on the removal of the cladding and conditions where transport limitations will allow for the removal of the products of the sublimation. A model also needs to be added to account for the decrease in grain size due to the sublimation of the lattice and the effect of this change in grain size on the diffusion process.

7.2.3 Open Porosity Processes

Once a fission product species is at the surface of the grains, in either a condensed or gaseous form, the extra-granular transport processes treated in Eq. (7.4) take over. In the current version of VICTORIA transport of gaseous species within the open porosity is by gas phase diffusion. Condensed species

in the open porosity are transported by a surface diffusion mechanism. In other geometric configurations either solid state or Knudsen diffusion mechanisms are operative. The physical state of a species is determined by chemical equilibrium considerations. The nature of the transport process depends on the geometry of the particular region being considered.

7.2.3.1 Chemical Equilibrium

Within the fuel open porosity region one can have significant concentration of fission product atoms which have migrated out of the fuel grains. Since this migration is a random process, these atoms are intimately mixed within the open porosity. Given this well-mixed volume with gaseous elements in intimate contact with those residing on the surface, the achievement of chemical equilibrium in this case is taken as a reasonable approximation.

7.2.3.2 Diffusive Transport

Transport of chemical species within the open porosity is treated directly by Eq.(7.4). The only means of transport currently allowed in this region is diffusion. The determination of the diffusion coefficient to be used in the solution of the governing equation is dependent on the chemical identity and physical phase of the transported species and on the geometry of the cell in question.

The geometry of the cell is used to determine whether the mode of diffusive transport is by Knudsen diffusion, by solid state diffusion, or by "open-space" diffusion.

A geometric configuration consisting of diffusive volumes with a characteristic dimension (say the pore radius) smaller than the mean free path of a gas phase molecule, at the temperature and pressure of the cell, puts the transport in a regime governed by Knudsen diffusion [7.8]. This mechanism has a diffusion coefficient for the i^{th} species of the form

$$D_i = \lambda \left[\frac{8kT}{\pi M_i} \right]^{1/2} \quad (7.7)$$

where λ is the pore diameter, M_i is the molecular weight of the i^{th} species and k is Boltzmann's constant.

A configuration in which intact solid is the transport medium requires a diffusion coefficient as given in Eq.(7.6). In this

case, values for D_0 and θ must be given. While both atomic and molecular species can, in principle, diffuse in this manner it must be recognized that molecules diffuse much more slowly than atoms in this regime.

Within the open porosity the most common type of diffusive transport is that identified as open-space transport. The form of the diffusion coefficient for this mechanism depends on the phase of the species in question.

For condensed phase species, diffusion is postulated to take place along the surfaces of the grains and the coefficient for the governing equation is given by

$$D = 40.e^{-54125/T} \text{ (m}^2\text{/sec)} \quad (7.8)$$

where T is the temperature (in Kelvin) of the region in question. This value of the diffusion coefficient was developed [7.9] for surface self-diffusion of UO_2 but is taken to be valid for all condensed phase species in the present version of VICTORIA.

For gas phase atoms and molecules a diffusion coefficient based on the Chapman-Enskog model [7.8] is used. The value of the coefficient is given by

$$D_i = 1.8583 \times 10^{-7} T^{3/2} \frac{[1/M_i + 1/M_m]^{1/2}}{P \sigma_{im}^2 \Omega_{im}} \quad (7.9)$$

where T (K) is the temperature, P (atm) is the pressure, M_i is the molecular weight of the diffusing species, M_m is the average molecular weight of the gas in the cell, σ_{im} is the Lennard-Jones interaction distance (in meters), and Ω_{im} is a dimensionless collision integral. In the current version of the code the values for σ_{im} and Ω_{im} are set in a data statement to an arbitrary average for small molecules at high temperatures. The value of M_m is contained in a data statement and is usually set for either steam, hydrogen, or a fixed ratio mixture of the two. The value of M_i is set in a data statement to a weighted average value for gaseous fission product species. A modification for determining the value of M_m and/or M_i during execution would be run-time consuming but not difficult to implement. The values of T and P are determined during execution. While it is true that all of these parameters affect the calculated diffusion coefficient, this effect is likely to be small compared to the uncertainty in the temperature and pressure dependence in this formulation. A

small error in the power dependence of either of these parameters will cause a large change in the diffusion coefficient for the temperatures and pressures operative in a severe reactor accident. The correct functional form of D_i in this regime of temperatures and pressures is not known. Further comparison of the code against experimental results may help to resolve this issue.

7.2.4 Ex-Pellet Processes

Once a fission product species reaches the surface of the fuel pellet stack (or an exposed debris bed surface, or other surface external to the fuel) one needs to consider transport through a different medium. With intact fuel there is a fuel-clad gap which provides a better medium for gas phase diffusion than the intergranular region but may not provide a suitable venue for surface diffusion. For other geometries there are obstructions such as crusts or structural materials, which present changes in transport properties and must be treated. One must also consider the interfaces involving fuel, gap, clad, the bulk gas region, and other materials that are present in the various geometries treated.

7.2.4.1 Equilibrium Chemistry

For any of the fuel configurations treated in VICTORIA the the ex-pellet volume contains at least one region that has a more open geometry than that interior to the pellets. This open geometry allows for a well-mixed gas phase above a condensed phase lying on one or more interfacial surfaces. Given the well-mixed nature of the gas phase in this region, the assumption of chemical equilibrium for this phase is a reasonable one. Further, the ready and open access of the gas phase to the condensed phase in this region makes the assumption of chemical equilibrium between the gas and condensed phase acceptable as well. There still remains the question of whether or not the condensed phase is a single equilibrium phase or if it consists of several independent phases, each composed of one or more species at equilibrium. The answer to this question depends on the composition and area of the surface and the rates of reactions and rates of transport of the various species in the condensed phase. At the present time the gaseous and condensed phase species are treated as one equilibrium system. The equilibrium calculation in this region includes both the gaseous species residing in the gap and the condensed species that are

found in the innermost clad cell. Once the position of equilibrium is determined, the species are partitioned such that gaseous species are only found in the gap and condensed species are placed in the innermost clad cell.

7.2.4.2 Transport

Fission product transport across the fuel clad gap region is treated as a purely diffusive process. The mathematical treatment is very similar to that used in the open porosity region. Gas phase species are treated as in the open porosity with the exception that the porosity of the gap is always taken to be unity. This can be altered by the user in the VIKFACE subroutine to account for a closed or partially closed gap. The diffusion coefficients used for the gas phase species are calculated in the same way as those used in the open porosity and are subject to the same reservations. The diffusion of condensed species across the gap is not allowed; for condensed species the diffusion coefficient is set to zero in the gap region. This is to account for the fact that there is no physical mechanism to allow a condensed species to migrate across an open space. There is no continuous surface for migration like that available in the open porosity. A surface diffusion mechanism could be added for a gap cell in which there is contact between the clad and the fuel pellets.

7.3 Bulk Gas Species Continuity

The conservation of gaseous species in the bulk gas flow within the reactor vessel is treated in this section. Although not explicitly indicated, the nominal functional dependence of all the dependent variables is on time and two-dimensional space. The relevant two-dimensional conservation equation is first presented in exact form, followed by the approximation of the partial derivatives with finite differences. This yields a rather large system of algebraic equations (one equation is solved for each species) that requires solution at every time-step. The section then concludes with some code user information.

7.3.1 Species Conservation Equation

The history of species j in the bulk gas is governed by the following Eulerian conservation equation written in two-dimensional cylindrical coordinates (r, z) :

$$\begin{aligned} \frac{\partial N_{gj}}{\partial t} dV = & -\frac{\partial}{\partial r} (\vec{n}_{rj} dA_r) dr - \frac{\partial}{\partial z} (\vec{n}_{zj} dA_z) dz - \frac{\partial}{\partial r} (\vec{n}_{rfj} dA_{fr}) dr_f \\ & - \frac{\partial}{\partial r} (\vec{n}_{rsj} dA_{sr}) dr_s \end{aligned} \quad (7.10)$$

where N_{gj} is the molar concentration of species j in the bulk gas. The quantity $\vec{n}_j$ (e.g. $\vec{n}_{rj}$, $\vec{n}_{sj}$) in Eq.(7.10) is defined as the total molar flux vector, which can be expressed in terms of the bulk gas velocity vector $\vec{v}$ and a relative diffusive flux term that is assumed to be described by Fick's law:

$$\vec{n}_j = N_{gj} \vec{v} - D_j \nabla N_{gj} \quad , \quad (7.11)$$

where D_j is the molecular diffusion coefficient for species j in the gas mixture the vectors $\vec{n}_{rfj}$ and $\vec{n}_{rsj}$ are the same as the $\vec{n}$ and pertain to fluxes to and from the fuel surface region and structure surface region, respectively.

The chemistry source appearing in Eq.(7.10) is next cast in terms of a delta-function adjustment made to the species initial condition at the beginning of each time-step:

$$S_{chmj} = \sum_{\alpha=1}^A \delta\{t - t^\alpha\} (N_{gj} - N_{gjo}) \quad , \quad (7.12)$$

where the Dirac delta function integrates to one at each time t^α in the following manner:

$$\int_{t^{\alpha+\epsilon}}^{t^{\alpha-\epsilon}} \delta\{t-t^\alpha\} dt = 1 \quad . \quad (7.13)$$

The intent behind the form of Eq.(7.13) is that the effect of an equilibrium chemistry calculation at the beginning of each VICTORIA time-step is modeled by discontinuously adjusting the concentration before chemistry (represented by N_{gjo}) to its new value after chemistry (represented by N_{gj}). After this adjustment of the species initial values, Eq.(7.10) is then integrated over the time-step in the usual finite difference fashion to be described below. The changes to the gas species concentrations described by Eq.(7.12) not only account for

chemical changes among gaseous species, but also includes molecular exchange with aerosol particles due to phase change and chemical reaction.

7.3.2 Numerical Scheme

The spatial partial derivatives appearing in Eq.(7.10) are approximated using first-order-accurate, donor-cell differences for the convective terms, and second-order accurate central differences for the diffusive terms. The donor-cell scheme is utilized here for MELPROG compatibility reasons. In particular, molar fluxes need to be defined at cell boundaries to conform to MELPROG and TRAC numerical conventions.

In the difference equations given below, the time (t) index is α , the r,z space coordinates indices are m,n respectively, and the notation $m+$, $n+$ or $m-$, $n-$ is shorthand for $m+1/2$, $n+1/2$ or $m-1/2$, $n-1/2$, respectively; the absence of a t, r, or z index implies the presence of an $\alpha+1$, m or n index respectively. The reader is referred to Fig. 7.6 for an illustration of the space discretization scheme.

The convective derivatives from Eq.(7.10) are approximated first as:

$$\begin{aligned} \frac{\partial}{\partial r}(uN_{gj}dA_r)dr &\sim A_{m+}u_{m+} [N_{gj}f_{m+} + N_{gjm+1}g_{m+}] \\ &- A_{m-}u_{m-} [N_{gjm-1}f_{m-} + N_{gj}g_{m-}] \end{aligned} \quad (7.14)$$

$$\begin{aligned} \frac{\partial}{\partial z}(vN_{gj}dA_z)dz &\sim A_{n+}v_{n+} [N_{gj}f_{n+} + N_{gjn+1}g_{n+}] \\ &- A_{n-}v_{n-} [N_{gjn-1}f_{n-} + N_{gj}g_{n-}] \end{aligned} \quad (7.15)$$

where the donor-cell parameters are defined as

$$f_{m\pm} = \begin{cases} 1 & \text{if } u_{\pm} \geq 0 \\ 0 & \text{if } u_{\pm} \leq 0 \end{cases} , \text{ and} \quad (7.16)$$

$$g_{m\pm} = 1 - f_{m\pm} . \quad (7.17)$$

The diffusive terms are next approximated with central differences:

$$\frac{\partial}{\partial r} \left(-D_j \frac{\partial N}{\partial r} g_j \right) dr \sim A_{m+}j_{rm+} - A_{m-}j_{rm-} , \quad (7.18)$$

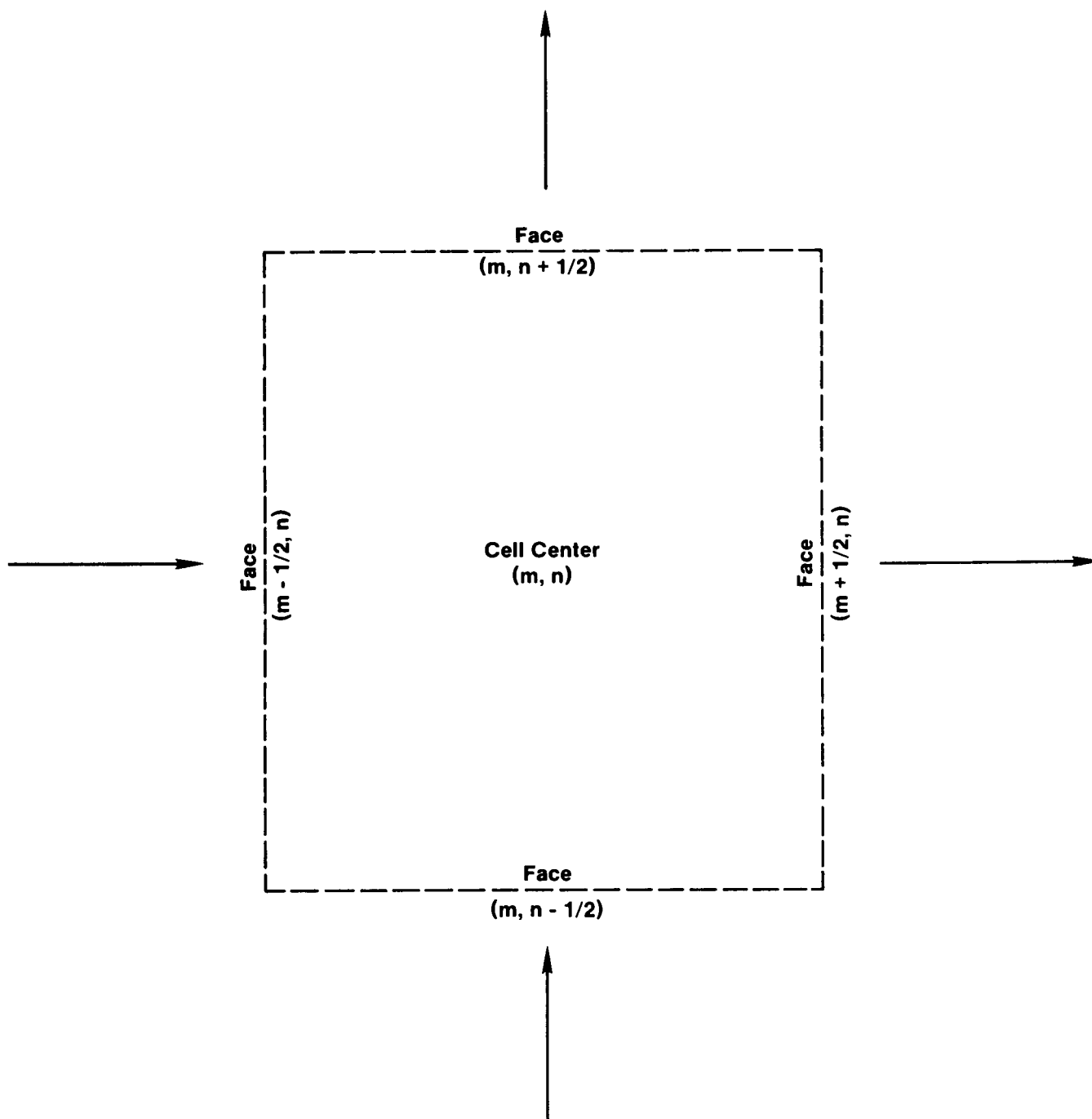


Fig. 7.6 Spatial indexing scheme shown for a single bulk gas cell.

$$\frac{\partial}{\partial z} \left(-D_j \frac{\partial N_{gj}}{\partial z} \right) dz \sim A_{n+} j_{zn+} - A_{n-} j_{zn-} , \quad (7.19)$$

where the diffusive fluxes (j 's are also central-differenced terms) are:

$$j_{rm+} = -D_{m+} \left[\frac{N_{gm+1} - N_{gm}}{r_{m+1} - r_m} \right] , \quad (7.20)$$

$$j_{rm-} = -D_{m-} \left[\frac{N_{gm} - N_{gm-1}}{r_m - r_{m-1}} \right] , \quad (7.21)$$

$$j_{zn+} = -D_{n+} \left[\frac{N_{gn+1} - N_{gn}}{z_{n+1} - z_n} \right] , \text{ and} \quad (7.22)$$

$$j_{zn-} = -D_{n-} \left[\frac{N_{gn} - N_{gn-1}}{z_n - z_{n-1}} \right] . \quad (7.23)$$

Lastly, the time derivative of Eq.(7.10) is backward differenced as

$$\frac{\partial N_{gj}}{\partial t} \sim \frac{N_{gj}^{\alpha+1} - N_{gj}^{\alpha}}{\Delta t} , \quad (7.24)$$

where $\alpha+1$ refers to the new time point and Δt is the integration time step. Note that diffusion coefficient in the above difference equations is defined at the boundary between cells. Since this quantity is typically calculated from cell-centered information (e.g., temperatures), a space-averaged value is needed. Use of the same boundary value for D_j by juxtapositioned cells is necessary to avoid a non-conservative scheme.

The substitution of the total flux (Eq.(7.11)), the sources (Eq.(7.12)), and the difference equations (Eqs.(7.14) through (7.24)) into Eq.(7.10) and approximating the differential volume dV with a finite cell volume ΔV produces a single algebraic equation containing the seven unknown species concentrations N_{gm} , N_{gm-1} , N_{gm+1} , N_{gn-1} and $N_{fuelsurface}$ and $N_{structure surface}$ for each spatial cell. Once appropriate boundary conditions are specified and since the concentrations are being solved for implicitly, the resulting system of equations can then be solved to give the concentration distribution over the entire space domain.

Note that the solution of the above system assumes that geometric, chemical, and transport coefficient information is known. For example, the diffusion coefficient appearing in Eq.(7.11) assumes the Chapman-Enskog formula [7.8], which requires temperature and pressure information for evaluation. In other words, temperature, pressure and velocity field information provided by some external means (external to VICTORIA) must be available before the bulk gas continuity problem can be solved.

At each time step then a seven-stripe matrix of equations must be solved for the species concentrations. Solution of this matrix in VICTORIA is accomplished by dividing the matrix into three tridiagonal matrices that are solved sequentially using a system subroutine.

Even though the transport terms and part of the convective source are treated implicitly in the above numerical scheme, the explicit nature (no instantaneous exchange of information) of the chemistry source will result in explicit oscillation of the species concentrations if the time step becomes too large.

7.3.3 Information for the Code User

The solution of the bulk gas species continuity problem is addressed in subroutine VGTRAN. This routine calls subroutine SGTSL to solve the tridiagonal matrices. The calculational flow of VGTRAN is illustrated by the flow chart in Fig. 7.7.

7.4 Chemistry

7.4.1 Introduction

Chemical interactions of fission products can have an important effect on their behavior, affecting the timing, chemical form, and amount which may be released from the vessel or RCS should either fail. Two areas of particular concern are: the chemical form of iodine, which affects its volatility; and, the importance of resuspension by reentrainment of deposited aerosols and by revaporization, particularly at the time of vessel or RCS failure [7.10].

Assessment of these concerns requires detailed consideration of chemical interactions in the fuel and gas regions and with structures. For example, microscopic studies [7.5] of LWR fuel

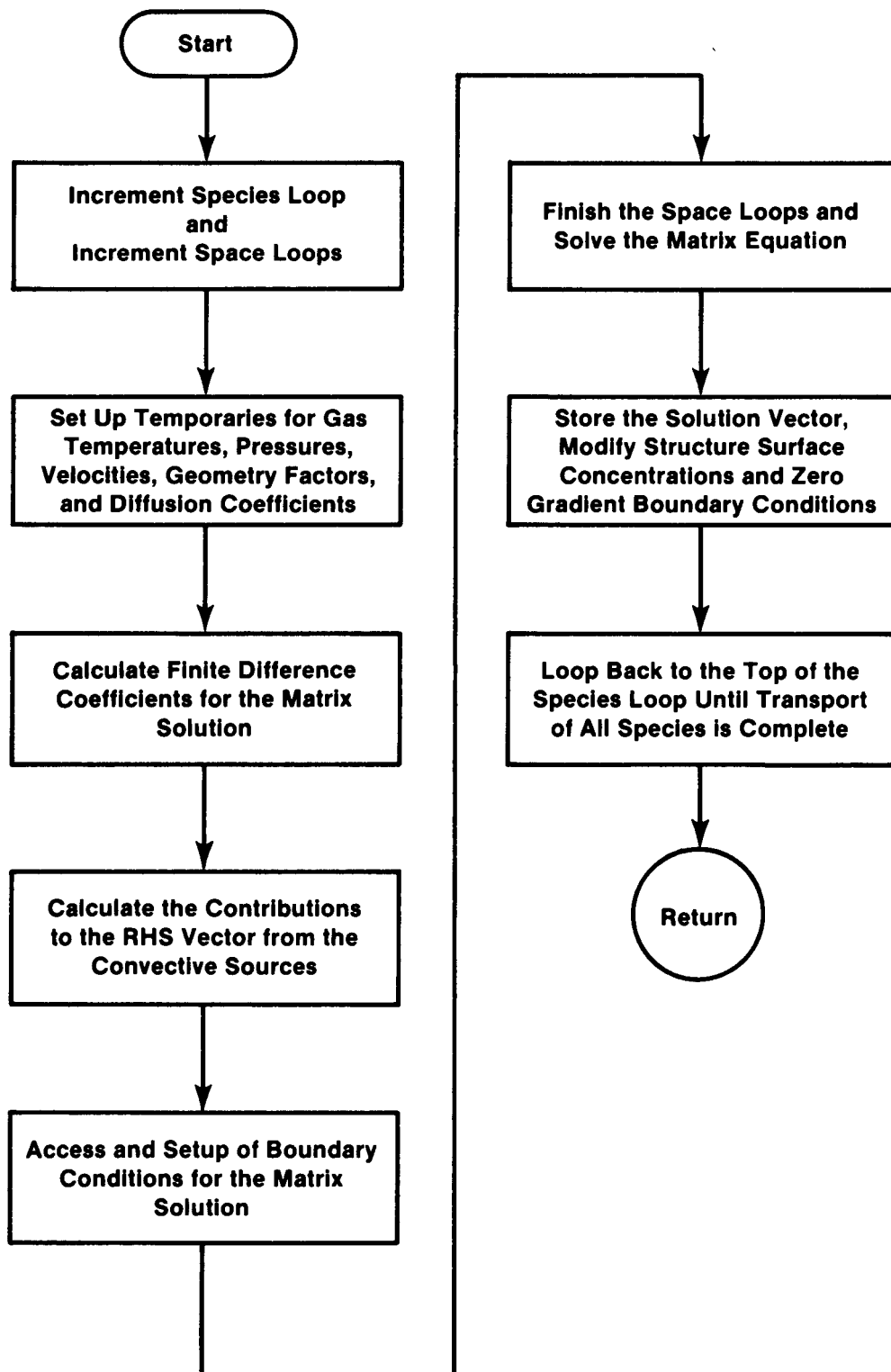


Fig. 7.7 Flowchart of the logic used in subroutine VGTRAN to solve the bulk gas species continuity equations.

have shown that fission product elements diffuse within the fuel grains to form associated species either as part of the UO_2 matrix or as separate phases at grain interstices. This affects fission product element concentrations within the grains and hence the driving force for diffusion to the grain edges and into the open porosity and their vapor pressures there.

Some fission products may condense on the clad to react with its constituents, e.g., Te favorably reacts with Zr and Sn [7.11]. Sn can vaporize to react with fission products in the vapor phase [7.12]. Both processes affect fission product diffusion between the fuel and the clad.

Clad oxidation by steam generates large amounts of heat. It affects the chemical behavior of fission products both through the altered temperature in the vicinity of the clad, and by reducing the oxygen chemical potential in the gas.

Fission products mix and react with steam, molecular hydrogen and vaporized structural constituents in the gas region. This affects the partition between condensed (aerosol) and gaseous species and physical characteristics of the aerosol which in turn affects transport mechanisms between the gas and structure surfaces.

Structural components are generally made of zircaloy, Inconel (an alloy consisting of 76% Ni, 16% Cr, and 8% Fe) [7.13] or stainless steel. Special components are also present, such as control rods in LWR's containing an Ag/In/Cd alloy and control blades in BWR's containing B_4C powder. Fission products and other gas-borne materials can condense or deposit onto structures and react amongst themselves and with structural constituents; trace constituents such as Si and Cr in steel may be important [7.14]. Such interactions affect removal rates from the gas and resuspension.

It is challenging to treat these processes, and others not mentioned, comprehensively and realistically. Some simplifying assumptions, indeed, must be made to avoid prohibitive computational labor. The main assumptions used are described below.

Any liquid water (i.e., $\text{H}_2\text{O}(c)$) present in the vessel after fission products are released from the fuel is ignored currently. Because VICTORIA is intended only to model unrecovered severe

accidents, i.e., those for which no emergency cooling water injection occurs during or after fuel failure, fission products are not expected to condense in any remaining water pools. Steam, on the other hand, may be present; therefore, chemical interactions involving steam are treated. Temperatures within the vessel are assumed to be too high for steam condensation to occur.

It is assumed that all chemical interactions are transport limited. Species diffusion is therefore treated as required and chemical equilibrium pertains in all regions. This has significant advantages over the alternative of treating chemical rate limiting processes: computation costs are less, many species and elements can be treated, and much more data are available.

All phases are assumed to be ideal. Data are not available to treat non-ideality comprehensively and chemical equilibrium is much more difficult to calculate for complex non-ideal systems. The error incurred in this assumption decreases with increasing temperature.

At most one condensed phase of each chemical substance is assumed to exist throughout the prevailing conditions of any modeled accident.

Transmutation and isotope effects are assumed to be unimportant. Consequently, element masses are conserved, relative isotopic abundances are everywhere constant (but not necessarily time independent) and radioactive decay can be independently calculated.

Some of these assumptions are evaluated in more detail later in this chapter. They are not evaluated here and their validity depends on the application.

Given these assumptions a unified approach to calculating chemical equilibria is derived. Its application to the gas and fuel regions and structure surfaces is then discussed. Careful attention is paid to how the approach might be extended to achieve greater reality in the treatment.

Third, the solution method of the equations which determine chemical equilibrium is discussed. Even with the simplifications outlined above these calculations can dominate the computational

costs of VICTORIA or VICTORIA coupled with MELPROG and TRAC. Considerable effort has therefore been expended to achieve maximum calculational efficiency. Standard methods have been found to be inefficient or, as discussed later, suffer numerical difficulties. New methods of solving the equations have therefore been developed. They have proved to be very efficient and are discussed in detail in view of their general applicability.

The section concludes with a discussion of the free energy data for the chemical species treated in the code.

7.4.2 Equilibrium Chemistry

7.4.2.1 Derivation of the equations

Conditions for chemical equilibrium. The condition for chemical equilibrium in a system at temperature, T , and pressure, P , is

$$\sum_i \mu_i dn_i = 0 \text{ and } n_i \geq 0, \quad (7.25)$$

where μ_i is the chemical potential of the i^{th} species and n_i is the corresponding number of moles. The sum is over all species, counting separately species with the same chemical formula in different phases. In this and subsequent formulae the limits of summation are not shown when the sum is over all values of the index for which the variables in the sum are defined.

The n_i are not independent when phase changes or chemical reactions occur; they are constrained by conservation of mass of the elements. Let the number of moles of each element be denoted by N_j for the j^{th} element, counting contributions from all species. Letting a_{ij} denote the number of molecules of the j^{th} element in the i^{th} species; then in different form we have for each element:

$$dN_j = \sum_i a_{ij} dn_i. \quad (7.26)$$

The method of Lagrangian multipliers can be used to find the solution to Eq.(7.25) subject to these constraints. The result is

$$\mu_i = \sum_j a_{ij} \lambda_j, \quad (7.27)$$

where λ_j is the Lagrangian multiplier corresponding to the j^{th} constraint.

There are n_s of these relations and n_e parameters, where n_s and n_e are the number of species and elements, respectively, so the λ_j can be eliminated to find $n_r = n_s - n_e$ relationships between the μ_i . There are a number of ways this can be done. The simplest arises when the elements are included in the sum over species (i.e. the unassociated elements are present in the system). Let the first n_e species be chosen to be these elements and denote this set by e , then $\lambda_j = \mu_j$, $j \in e$, and

$$\mu_i = \sum_j a_{ij} \mu_j . \quad (7.28)$$

Only the last n_r of these equations are useful; the first n_e are identities.

More generally, we can usually choose n_e species chemical potentials to eliminate the λ_j . Let the set of indices be denoted by d . If the determinant of the n_e rows of a_{lj} for $l \in d$ is non-zero then A_{jl} for $j \in e$ and $l \in d$, the inverse matrix, exists and can be used to invert Eq.(7.27). We find

$$\lambda_j = \sum_l A_{jl} \mu_l . \quad (7.29)$$

Now we can eliminate the λ_j from Eq.(7.27), giving

$$\mu_i = \sum_{j,l} a_{ij} A_{jl} \mu_l . \quad (7.30)$$

Again, only the last n_r of these equations are useful.

The requirement that A_{jl} exists means that we cannot choose any set of n_e chemical potentials to eliminate the λ_j . Indeed, we must choose a set whose corresponding species are linearly independent in the sense that all other species can be made from linear combinations of them. This can always be done when $\text{rank}(a_{ij}) = n_e$.

The usual form in which the conditions for chemical equilibrium are expressed is

$$\sum_i \nu_{ki} \mu_i = 0 . \quad (7.31)$$

The ν_{ki} $k = 1, \dots, n_r$ are the coefficients in the k^{th} chemical reaction, expressed as

$$\sum_i \nu_{ki} S_i \neq 0 , \quad (7.32)$$

where S_i denotes the chemical formula of the i^{th} species. An equivalent formulation to Eq.(7.31) is

$$\sum_{m,i} T_{km} \delta(m, i-n_e) \{ \mu_i - \sum_{j,l} a_{ij} A_{jl} \mu_l \} = 0, \quad (7.33)$$

where T_{km} $k, m = 1 \dots n_r$ is of rank n_r and $\delta(m, i-n_e)$ is the Kronecker delta which is zero unless $m = i-n_e$, when it is unity. It is seen that Eq.(7.30) corresponds to a special choice of the n_r chemical reactions: i.e., T_{km} is the unit matrix. Later it will be shown that the choice of Eq.(7.33) has the advantage that it simply leads to a reduction in the number of unknowns. This is less straightforward with the formulation used in Eq.(7.31).

Equations for two phases at given T and V. There are n_e unknowns, μ_j for $j \in e$, and n_e constraints, for the conservation equations for N_j . So, in principle, the equations are closed once the functional dependence of the μ_i on the n_i are known. Further, progress cannot be made without making the dependence explicit.

Henceforward, it will be assumed that the separate phases in the system are ideal. There are two practical reasons for this. First, sufficient data does not exist to treat, in general, the non-ideality in the systems considered (though models exist which may be suitable). Second, the equations for non-ideal systems are much more difficult and (more to the point) expensive to solve.

Consider first a system (for each cell) with just two phases, one condensed (either solid or liquid) and the other gaseous. Further suppose that the temperature and total volume, V , of the system are specified prior to each equilibrium calculation; these are the present assumptions in VICTORIA in all regions for which equilibrium chemistry is assumed.

The chemical potentials of the gaseous species are

$$\mu_i = \mu_i^0(P_0, T) + RT \log_e (P_i/P_0), \quad i \in g, \quad (7.34)$$

where μ_i^0 is the standard Gibbs free energy of the pure i^{th} gaseous species; P_0 is typically atmospheric pressure and μ_i^0 is otherwise a function of T only. R is the universal gas constant, P_i is the species partial pressure and g denotes the set of indices which correspond to gaseous species.

It is convenient to rewrite this equation. Define the (pseudo) concentration of each gaseous species by $c_i = n_i/V$ for

all $i \in g$. Then $P_i = c_i RTV/V_g$, where V_g is the volume of the gaseous phase, and

$$\mu_i = \mu_i^{\circ}(P_o, T) + RT \log_e (c_i RTV/V_g P_o), \quad i \in g. \quad (7.35)$$

The chemical potentials of the condensed species are

$$\mu_i = \mu_i^{\circ}(P_o, T) + RT \log_e (x_i), \quad i \in c, \quad (7.36)$$

where μ_i° is the standard Gibbs free energy of the pure i^{th} condensed species (specified as a liquid or solid), x_i is the mole fraction of the species in the condensed phase and c denotes the set of indices which correspond to condensed species. If c_i is defined by $c_i = x_i n_c / V$ for all $i \in c$, where n_c is $\sum_{i \in c} n_i$, then

$$\mu_i = \mu_i^{\circ}(P_o, T) + RT \log_e (c_i V / n_c), \quad i \in c. \quad (7.37)$$

Substitute Eq. (7.35) and (7.37) into Eq. (7.28), where the independent chemical potentials in Eq. (7.28) are chosen to be those for the gaseous elements, to find

$$c_i = R_i \prod_j c_j^{a_{ij}} \{V/V_g\}^{-1} \prod_k \{V/V_g\}^{a_{ik}}, \quad i \in g, \quad (7.38)$$

and

$$c_i = R_i \prod_j c_j^{a_{ij}} n_c / n_c^{\circ} \prod_k \{V/V_g\}^{a_{ik}} \quad i \in c. \quad (7.39)$$

R_i is the equilibrium constant for dissociation to form the gaseous elements, defined by

$$R_i = \exp(-\Delta G_i / RT) \{RT/P_o\}^{-1} \prod_j \{RT/P_o\}^{a_{ij}}, \quad i \in g, \quad j \in e. \quad (7.40)$$

and

$$R_i = \exp(-\Delta G_i / RT) \{n_c^{\circ} / V\}^{-1} \prod_j \{RT/P_o\}^{a_{ij}}, \quad i \in c, \quad j \in e. \quad (7.41)$$

n_c° is a reference value of n_c , whose introduction will become apparent shortly. ΔG_i is the relative standard species Gibbs free energy (relative to the gaseous elements), defined by

$$\Delta G_i = \mu_i^0 - \sum_j a_{ij} \mu_j^0. \quad (7.42)$$

Notice that the equilibrium constants have been defined so that they are independent of n_i ; they need only be calculated once each time a chemical equilibrium calculation is done since T and V are supposed to be fixed at the start of the calculation.

A closed set of equations can now be formed. Start with the mass balance Eq.(7.26) and divide this through by V to obtain

$$C_j = \sum_i a_{ij} c_i, \quad (7.43)$$

where $C_j = N_j/V$ for all $j \in e$. Now substitute Eq.(7.38) and (7.39) into this to find

$$C_1 = \sum_{i \in g} a_{i1} R_i \prod_j c_j^{a_{ij}} \{V/V_g\}^{-1} \prod_k \{V/V_g\}^{a_{ik}} \\ + \sum_{i \in c} a_{i1} R_i \prod_j c_j^{a_{ij}} n_c/n_c^0 \prod_k \{V/V_g\}^{a_{ik}}. \quad (7.44)$$

V_g is $V - V_c$ where V_c is the volume of the condensed phase which is a function of n_i for all $i \in c$. So, the unknowns in these equations are c_j for $j \in e$, V_g and n_c . n_c is determined by the requirement that the sum of the mole fractions in the condensed phase is unity:

$$n_c = V \sum_{i \in c} c_i. \quad (7.45)$$

The solution must satisfy $c_i \geq 0$ for all i . Either one or two solutions exist. One solution always consists of a gaseous phase and no condensed phase. If a second solution exists, then this will consist of two phases; the gas phase in the first solution must then have been supersaturated.

Two Approximations. No approximations have been made in the formulation of these equations. Two will now be made. The aim is to minimize the computational cost of obtaining their solution of the equations.

First, n_c^0 is chosen to have the value that n_c had at the end of the previous time step and n_c in Eq.(7.44) is approximated as n_c^0 . Equation (7.45) is used to update n_c after the other equations have been solved. We require $|n_c - n_c^0|/n_c \ll 1$, which can

be achieved if the time step is made sufficiently small. Notice that this approximation implies that the condensed phase can never vanish if it existed initially, though n_c may become very small after many time steps. The approximation breaks down when n_c^0 is zero initially.

Second, V_g is approximated as V . This implies that $|V - V_g|/V \ll 1$.

It is important to note that neither of these approximations leads to a mass balance error (they amount to making approximations in Eqs. (7.38) and (7.39) only). They affect only the equilibria among the species and hence alter only the relative proportions of species.

Equations (7.44) then become

$$C_i = \sum_j a_{ij} R_j \prod_k c_k^{a_{ik}} \quad (7.46)$$

which is a closed set of equations.

Since V , T and N_j have been specified, P is determined by the solution. It is given by

$$P = \sum_{i \in g} c_i R T V / V_g = \sum_{i \in g} c_i R T \quad (7.47)$$

This has implications for the application of these equations to VICTORIA which are discussed below.

V may change from one time step to the next. This affects the calculation of C_j . Let V^0 and C_j^0 denote their values at the end of the preceding time step. C_j^0 generally will have been modified from its value found from the equilibrium calculations at that time step by transport into and out of the region considered. To conserve mass C_j is found according to

$$C_j V = C_j^0 V^0 \quad (7.48)$$

7.4.2.2 Application to the Gas Region

This is the space between structures and fuel containing gas and aerosol; species immediately adjacent to or condensed onto fuel or structure surfaces are excluded. It is divided-up into a number of cells. Each is assumed to be well mixed; T , P and

gaseous species concentrations are therefore taken to be spatially constant inside. Further, a number of assumptions are made with regard to the chemistry calculations done in these cells. These will be discussed in turn.

The cells are assumed to be materially isolated during the chemistry step. This appears unrealistic at first sight because the cells are, in fact, interconnected. It is justified by the overall numerical approach in VICTORIA. In this approach the governing equations are split into convenient parts and each part is solved in turn, uncoupled from all others during execution of its solution. Coupling between the parts occurs after having solved one equation and before starting the solution of the next. The method is called 'splitting' or 'the fractional step method.' [7.15] Here the parts are defined so that transport of material into or out of the cells is not done during the chemistry step.

It is easy to show that the fractional step method is accurate to order Δt , where Δt is the time step, so errors can be made arbitrarily small by choosing Δt small enough. It is assumed that Δt is so chosen; its value will depend on the problem in hand so calculations should be done in which Δt is varied to test this. Experience with the code to date indicates that the material Courant condition is more stringent than this accuracy condition.

One other external constraint is required. At first sight, a reasonable choice would be that the cells are thermally isolated, i.e., $\Delta Q = 0$, where ΔQ is the amount of heat which enters the cell. Since V is constant, this is equivalent to $\Delta U = 0$, where U is the internal energy of the cell contents. ΔU is U after the chemistry step minus U just before the chemistry step. U is easily found from the species chemical potentials, with the result

$$U = \sum_i n_i \left\{ \mu_i^0 - T \frac{d\mu_i^0}{dT} \right\} - PV . \quad (7.49)$$

This applies to a mixture with uniform temperature which need not be in chemical equilibrium.

Instead, the calculation is done at constant T . Heat therefore flows into or out of the cell; the amount can be calculated by using Eq.(7.49). This alternative does not necessarily amount to an approximation; it affects the size of Δt

required for convergence and the coupling of the parts in the fractional step method; ΔU should be accounted for in the thermal hydraulics and heat transfer calculations. In fact, ΔU is ignored in calculations done with VICTORIA on its own because the thermal hydraulic calculations, etc., are done in advance. It is therefore assumed that ΔU is negligible compared to other sources of heat.

It was noted earlier that P is determined by the solution to the equations and, therefore, generally has a value different from that immediately before the chemistry step. This change in principle should be fed back to other parts of the calculation. Among those parts affected are the thermal hydraulics calculations which, as just noted, are done in advance. It is therefore assumed that the effect of the change in P is negligible.

Lastly, consider the underlying assumption that the cell contents are in chemical equilibrium. It is convenient because only one equilibrium calculation is required per cell. Two aspects, however, require further consideration.

First, since the gas is in equilibrium with the condensate, this assumption undermines the possibility of treating aerosol evaporation and condensation dynamically. It is therefore difficult to evaluate whether dynamic effects are important (note that a consequence of this assumption is that aerosol compositions do not vary with size), and a degree of arbitrariness enters into how the aerosol mass distribution should be updated due to the formation or removal of condensed species during the chemistry step.

These problems can be overcome by relaxing the chemical equilibrium assumption. For example, one might assume that the gaseous species are in mutual chemical equilibrium and the condensed species are in mutual equilibrium, though with particle compositions and equilibria depending on their size. This requires an equilibrium calculation to be done for each size group.

Second, mutual chemical equilibrium of the condensate (whether or not it is in equilibrium with the gas) is a diffusion limited process which, even for sub-micron particles, has a time constant large compared to typical time steps in VICTORIA. It may, therefore, be more realistic to assume that the condensed

species do not interact with one another, i.e., they are effectively immiscible. This is not amenable to a simple treatment if equilibrium between the gas and condensate is maintained, since the Gibbs phase rule limits the number of condensed species to no more than n_c at given T . However, it is amenable to a dynamic treatment of condensation and evaporation for which only an equilibrium calculation for the gas is required.

7.4.2.3 Application to the Structure Surface Region

Species can be transported to or from structure surfaces by turbulent and molecular diffusion in the gaseous phase and by aerosol deposition in the condensed phase.

Diffusion rates depend on vapor pressure differences between the bulk flow and the structure surfaces, and condensed species at the surfaces can evaporate and be reentrained into the bulk flow. A treatment of surface chemical effects is therefore important to the calculation of transport rates to and from surfaces.

A thin, constant thickness, region which extends over each structure surface is defined. Each region contains all gaseous and condensed material transported to the surface and is the source of vapor reentrained into the bulk flow. It can also contain some of the structure material (a few μm layer is typical) to enable chemical interactions with the surface to be treated. Chemical equilibrium is assumed in this region.

A number of assumptions are made which require evaluation. These are discussed in turn.

The temperature of the surface region is assumed to equal the structure surface temperature. Heat, therefore, flows into or out of the region during the chemistry step. It is assumed to be negligible compared to decay heating at the surface and heat transferred between the surface and other regions, e.g., the bulk gas and the structure interior.

The volume of the region, V , during the chemistry step is constant. It must be sufficiently small compared to the volume of the bulk gas region that one can justify that the gas within the surface region is in equilibrium with the condensed solid there. If it is assumed that transport between the gas and solid is the

rate limiting process then it is required that gas concentration differences across the thickness of the gaseous part of the surface region, Δh , are small compared to gas concentration differences between the surface and the bulk gas regions. The former can be found from standard wall profiles for smooth surfaces and the latter can be found from mass transfer correlations for the bulk flow. Generally for fully turbulent bulk flow one finds that Δh must be less than or of order of the thickness of the laminar sublayer.

V must also be large enough that $V \simeq V_g$ is valid, i.e., $V_c \ll V_g$ which, in view of the discussion above, implies that the thickness of the condensed layer must be much smaller than the thickness of the laminar sub-layer.

Evaluation of these requirements will depend on the application in hand, but notice that they become conflicting in the limit of increasing amounts of condensed surface deposits.

Lastly, consider the assumption that the condensate is in mutual equilibrium. As noted earlier, it may not be reasonable to assume this regardless of any interaction with the surface medium because of the limiting effect of diffusion in the condensed phase. Further, including the surface medium in the chemical equilibrium does not take account of surface inhomogeneities such as the presence of an oxide layer nor allow for the transient diffusion of species into the surface medium. And so, chemisorption is not properly treated.

Again, one can alternatively assume that the condensing species do not interact with one another. It may then be possible to treat their interaction with the surface medium dynamically. It follows from this that no chemical equilibrium calculations are required at the surface, with the possible exception of treating individually the equilibria of condensing species with the surface medium.

7.4.2.4 Application to the Clad/Gap Interface

The clad/gap interface is part of the fuel region which is discussed in detail in Section 7.2.

7.4.2.5 Application to the Fuel

This is also part of the fuel region and discussed in detail in Section 7.2.

7.4.2.6 The Solution Method

Change of Variables. Equations (7.46) are a set of n_e non-linear algebraic equations for the n_e unknowns c_j for $j \in e$. Although approximations have been made in their derivation for the two phase system, they are exact for single phase gaseous systems at constant V and so the following discussion has general applicability.

Consider the variation problem

$$\delta\Gamma = 0 \text{ and } c_j \geq 0 \text{ for } j \in e, \quad (7.50a)$$

where Γ is a function of the c_j for $j \in e$ given by

$$\Gamma = \sum_i c_i - \sum_j C_j \log_e(c_j), \quad (7.50b)$$

where the c_i are given by

$$c_i = R_i \prod_j c_j^{a_{ij}}, \quad (7.50c)$$

and $C_j \geq 0$ and $R_i > 0$.

These equations are solved by exploiting the existence of dominant species [7.16]. The equations are formulated in terms of new independent variables which are taken to be the set of c_i 's corresponding to any set of n_e independent species (denoted by d).

Define $\tilde{g}_1(d)$ as $\partial\Gamma/\partial c_1$ for $1 \in d$ and Γ expressed as a function of the new variables. The solution to Eq.(7.62) is

$$\tilde{g}_1(d) = 0. \quad (7.51)$$

Defining $g_1(d)$ to be $c_1 \tilde{g}_1(d)$ then

$$g_1(d) = 0. \quad (7.52)$$

$g_1(d)$ is given by

$$g_1(d) = \sum_j A_{j1} g_j, \quad (7.53)$$

where A_{j1} , $1 \in d$, is the matrix defined earlier (see Eq.(7.29)). A_{j1} exists, which is consistent with the definition of d .

Define new Jacobians $\tilde{J}_{lm}(d)$ and $J_{lm}(d)$, $l, m \in d$ in the new variables. They are given by

$$J_{lm}(d) = \sum_{j,k} J_{jk} Q_{jl} A_{km} , \quad (7.54)$$

and

$$c_l c_m \tilde{J}_{lm}(d) = J_{lm}(d) - \delta(l,m) g_m(d) . \quad (7.55)$$

$J_{lm}(d)$ is clearly positive definite whenever J_{jk} is positive definite. i.e. it is positive definite everywhere. $\tilde{J}_{lm}(d)$ is positive definite near the solution.

It is now valuable to express $g_l(d)$ and $J_{lm}(d)$ in terms of the c_i 's.

$$g_l(d) = c_l - \sum_j C_j A_{jl} + \sum_{i \in \bar{d}} \left(\sum_j a_{ij} A_{jl} \right) c_i \quad (7.56)$$

and

$$J_{lm}(d) = \delta(l,m) c_m + \sum_{i \in \bar{d}} \left(\sum_j a_{ij} A_{jl} \right) \left(\sum_k a_{ik} A_{km} \right) c_i , \quad (7.57)$$

where $\bar{d}$ denotes the set of species indices not contained in d .

It can be seen that if $c_m \gg c_i$ for $m \in d$ and $i \in \bar{d}$ then $J_{lm}(d)$ is nearly diagonal and is numerically well conditioned.

A suitable choice for d is the set of n_e dominant independent species defined as follows. Start with d consisting of the species with largest c_i for $i \in s$, where s is the set of all species. Add to d the species with largest c_i for $i \in s$ which is independent of those currently in d and repeat until d contains n_e species; d henceforward stands for this set of species unless stated otherwise. It is easily shown that this definition results in $c_m > c_i$ for $m \in d$ and all $i \in \bar{d}$ for which $\sum_k a_{ik} A_{km}$ is nonzero.

Solution Algorithms. Suppose d is known. Equation (7.56) can be used as the basis of a simple iterative procedure as follows. Choose initial values for c_i $i \in \bar{d}$ and substitute these into Eq.(7.56). Requiring $g_l(d)$ to be zero immediately gives estimates for the c_l , $l \in d$. These are used to up-date the c_i $i \in \bar{d}$ by, first, inverting Eq.(7.50c) to obtain the gaseous element concentrations

$$c_j = \prod_1 (c_1/R_1)^{A_{j1}} \quad (7.58)$$

and, then, substituting these into Eq.(7.50c) to obtain the remainder of the c_i , $i \in d$. This procedure is iterated by substituting the revised c_i , $i \in d$ into Eq.(7.56) and repeating the steps above.

This can be a very efficient algorithm. It has been used for the fuel region and it was found that only four iterations were required to obtain a relative accuracy in the molar concentrations of 1 part in 10^6 . The mass balances are exact if the iteration is terminated before the c_i , $i \in d$ are up-dated (whether or not the iteration has converged).

However, the algorithm suffers two disadvantages. First, it breaks down if any c_l , $l \in d$ becomes negative. This may happen if the initial guess is far from the solution or the wrong choice is made for d . This is not too serious a difficulty because one can constrain successive iterates of c_l , $l \in d$ to differ by no more than a positive constant factor. Second, d may not be known in advance, e.g., because it changes with time. The algorithm is unlikely to converge unless d has been correctly chosen. This suggests that this algorithm is not generally of use unless some other algorithm can be used to obtain an iterate sufficiently close to the solution for d to be calculated.

Equations 7.56 are nearly linear and $J_{lm}(d)$ is nearly diagonal. This suggests that a cyclically ordered Newton method, taking one equation at a time, will be effective.

Consider application of the Newton method to all the equations simultaneously. Because $J_{lm}(d)$ is positive definite, application to Eq.(7.56) gives

$$s_1 = - \sum_m J_{lm}^{-1}(d) g_m(d) . \quad (7.59)$$

Cholesky decomposition [7.17] can be used to find s_1 ; this method is faster and more stable numerically than Gaussian elimination.

Convergence Criteria. The residual element abundances are zero at the exact solution so the absolute magnitude of the residuals normalized by the element abundances, E_j , $j \in e$, are a relative measure of how well the governing equations are satisfied, i.e., we require

$$E_j = |g_j|/G_j \ll 1. \quad (7.60)$$

A relationship between these errors and the relative errors in the gaseous element molar concentrations can be found by doing a Taylor expansion about the c_j , giving

$$E_j \approx \left| \sum_k \left\{ \frac{\sum_i a_{ij} a_{ik} c_i}{\sum_i a_{ik} c_i} \right\} \frac{\delta c_k}{c_k} \right|, \quad (7.61)$$

where δc_k , $k \in e$, is the error in c_k ($\delta c_k = c_k(\text{exact}) - c_k$).

The quantity in brackets is of order unity or zero; therefore, $E_j \leq \max_{k \in e} (|\delta c_k/c_k|)$. E_j can be substantially smaller than this limit when J_{jk} is ill-conditioned. From Eq. (7.50c) it is found that $\max_{k \in e} (|\delta c_k/c_k|) \approx \max_{l \in d} (|\delta c_l/c_l|)$ and so $E_j \leq \max_{l \in d} (|\delta c_l/c_l|)$. Estimate $\delta c_l/c_l$ as s_l and require $\max_{l \in d} (|s_l|) \leq \sigma$ where $\sigma \ll 1$. Typically σ is chosen to be 10^{-6} . If s_l is a valid estimate of $\delta c_l/c_l$, then $\max_j (E_j) \leq \sigma$; however, s_l can be significantly smaller than $\delta c_l/c_l$, for example, when many small steps are taken in the direction of the solution.

7.4.2.7 Subroutine Design

The subroutines for calculating R_i and solving the equations using the methods outlined in the previous subsection (CHEMRS, FUELSOLV and GASSOLV, respectively) are straightforward to design; clearly it is very important that the solution method should take as few iterations as possible. However, the computing time of each iteration step can be dominated by the calculation of c_i , $i \in s$, from c_j , $j \in e$, and g_j and J_{jk} from c_i , $i \in s$ when n_s and n_e are large. It is therefore equally important that the subroutines which do these calculations (CHEMWS, CHEMGE and CHEMHE respectively) should be as efficient as possible. Their design is less straightforward and is discussed here.

Consider c_i , $i \in s$, first. They are calculated from Eq. (7.50c). If they are calculated within loops over the indices i and j then the number of multiplications required to calculate all the c_i excluding multiplications required for the exponentiations is $n_s \times n_s$. With $n_s = 14$ and $n_e = 99$ (the values currently in VICTORIA) this number is 1386. If the exponentiations are calculated in advance by multiplication then roughly an additional 100 multiplications are required. Economies can be made by vectorizing the loop over species but at the expense of

overhead associated with the vectorization, having to do the exponentiations within the loop, and not being able to exploit to the full the possibilities that a_{ij} is sparse (as is generally the case), some of the c_j , $j \in e$, are zero or only one or a few of them have been revised (i.e., only some of the c_i require revision), as could be the case for the cyclic Newton method.

The Jacobian matrix is symmetric and many components are zero due to the sparseness of a_{ij} . At most 32 components in the lower triangle of 91 components are nonzero. It is also found that the number of nonzero terms contributing to each component is typically small compared to 99 and that the minimum number of multiplications required to calculate each component compared to 198 is also small, indeed, it is often zero.

It is advantageous to calculate c_i , g_j and J_{jk} without loops over the indices. This has been confirmed in computational studies which show that an overall factor of five reduction in computing time is achieved.

7.4.3 Species

In all regions the basic chemistry calculation performed is a determination of chemical equilibrium based on a minimization of the free energy of the system. The current basis of the calculation is a set of 99 species made up from 14 different elements. The species are listed in Table 7.1 and the elements in Table 7.2.

The free energy data used in the code is given in Table 7.3 for each of the 99 species. The data is in the form

$$G = A + BT + CT^2 + DT^3 \quad (7.62)$$

for each species and G is in kilocalories/mole. The zero of energy used in this formulation of the free energies is that the enthalpy of the elements in their standard form at 298.15 K is zero. This choice of reference states allows phase change for both the elements and more complex chemical species to be treated in a continuous manner.

The method for obtaining the constants for each species in the above equation is as follows. The enthalpy and entropy at typically 1000 K and heat capacity data as a function of temperature above 1000 K for each species was taken from the

source listed in the reference column. This data was then used to construct a table of the free energy as a function of temperature at 20 degree intervals from 1000 K to 3000 K. The table for the condensed phase species uses data for the solid below the melting point and for the liquid above the melting point. This table for each species is then fitted with a third order polynomial using a least squares fitting routine. A random comparison of tabulated free energies taken from Refs. [7.18] and [7.19] with values calculated using the polynomial for each species shows agreement to within $\pm 5\%$.

Several significant elements are missing from the present list. Those currently believed to be most important are shown in Table 7.4. The addition (or deletion) of an element requires a larger change in the coding than that required for species composed of the current element set. The next revision to the data set is planned to include the elements in Table 7.3 and the necessary species additions and deletions.

Table 7.1		
VICTORIA SPECIES SET		
<u>Molecule</u>	<u>Gaseous</u>	<u>Condensed</u>
CsI	X	X
CsOH	X	X
(CsOH) ₂	X	
Cs ₂ O	X	X
Cs	X	X
Cs ₂ UO ₄		X
Cs ₂ U ₂ O ₇		X
Cs ₂ MoO ₄		X
BaOH	X	
Ba(OH) ₂	X	X
BaO	X	X
Ba	X	X
BaTe	X	X
BaI	X	
BaI ₂	X	X
BaZrO ₃		X
BaUO ₄		X
BaMoO ₄		X
I	X	
I ₂	X	
HI	X	
H	X	
H ₂	X	
H ₂ O	X	
Sr	X	X

Table 7.1 (cont.)		
VICTORIA SPECIES SET		
<u>Molecule</u>	<u>Gaseous</u>	<u>Condensed</u>
SrO	X	X
SrOH	X	
Sr(OH) ₂	X	X
SrI	X	
SrI ₂	X	X
SrZrO ₃		X
SrUO ₃		X
SrMoO ₄		X
Zr	X	X
ZrO	X	
ZrO ₂	X	X
Sn	X	X
SnO	X	X
SnO ₂	X	X
SnI ₂	X	X
SnTe	X	X
SnH ₄	X	
Te	X	X
Te ₂	X	
TeO	X	
TeO ₂	X	X
Te ₂ O ₂	X	
H ₂ Te ₂	X	
U	X	X
UO ₂	X	X
UO ₃		X
U ₃ O ₈		X
U ₄ O ₉		X
Kr	X	
Xe	X	
Fe	X	X
FeO	X	X
Fe ₂ O ₃		X
Fe ₃ O ₄		X
FeI ₂	X	X
Fe ₂ I ₄	X	
FeTe		X
FeTe ₂		X
Mo	X	X
MoO	X	
MoO ₂	X	X
MoO ₃	X	X
O ₂	X	
O	X	

Table 7.2	
VICTORIA ELEMENT SET	
Cesium	Uranium
Iodine	Zirconium
Barium	Tin
Strontium	Iron
Xenon	Hydrogen
Krypton	Oxygen
Tellurium	Molybdenum

Table 7.3					
FREE ENERGY DATA FOR THE 99 SPECIES IN VICTORIA					
	<u>A</u>	<u>B</u>	<u>C</u>	<u>D</u>	<u>Ref.</u>
'csic'	-0.7364e+02	-0.3589e-01	-0.8181e-05	-0.1583e-09	7.18
'csig'	-0.2524e+02	-0.7401e-01	-0.4486e-05	0.3432e-09	7.18
'csohc'	-0.9010e+02	-0.3635e-01	-0.9405e-05	0.7195e-09	7.19
'csohg'	-0.5792e+02	-0.6517e-01	-0.6490e-05	0.4750e-09	7.19
'csg'	0.2032e+02	-0.4407e-01	-0.2334e-05	0.1679e-09	7.18
'cs2og'	-0.1720e+02	-0.8144e-01	-0.6693e-05	0.5114e-09	7.19
'cs2oh2g'	-0.1579e+03	-0.9141e-01	-0.1342e-04	0.9649e-09	7.19
'csc'	0.3132e+01	-0.2506e-01	-0.3567e-05	0.2729e-09	7.19
'cs2oc'	-0.7053e+02	-0.3476e-01	-0.1177e-04	0.5941e-09	7.19
'cs2uo4c'	-0.4405e+03	-0.6580e-01	-0.2095e-04	0.1444e-08	7.20
'cs2u2o7c'	-0.7134e+03	-0.1411e+01	-0.3760e-0	0.2589e-08	7.21
'cs2moo4c'	-0.3506e+03	-0.6815e-01	-0.2469e-04	0.1349e-08	7.22
'baohg'	-0.5014e+02	-0.6450e-01	-0.6352e-05	0.4498e-09	7.19
'baoh2g'	-0.1433e+03	-0.8161e-01	-0.1101e-04	0.7971e-09	7.18
'baoh2c'	-0.2132e+03	-0.3567e-01	-0.1625e-04	0.1244e-09	7.18
'baog'	-0.2645e+02	-0.5981e-01	-0.4047e-05	0.2781e-09	7.19
'baoc'	-0.1353e+03	-0.1093e-01	-0.1016e-04	0.6290e-09	7.18
'baig'	-0.9134e+01	-0.6698e-01	-0.4368e-05	-0.7546e-09	7.19
'bai2g'	-0.1274e+03	-0.6142e-01	-0.6704e-05	0.5129e-09	7.19
'bai2c'	-0.1317e+03	-0.5118e-01	-0.1302e-04	0.9963e-09	7.18
'bag'	0.4581e+02	-0.4400e-01	-0.1430e-05	-0.6106e-10	7.18
'bac'	0.5148e+01	-0.2015e-01	-0.4679e-05	0.3580e-09	7.18
'batec'	-0.6086e+02	-0.2742e-01	-0.5920e-05	0.3764e-09	7.18
'bateg'	4.7990e+01	-7.6300e-02	1.9300e-07	2.5500e-11	*
'bazro3c'	-0.4141e+03	-0.3546e-01	-0.1507e-04	0.1068e-08	7.18
'bauo4c'	-0.4654e+03	-0.4943e-01	-0.1882e-04	0.1072e-08	7.23
'bamoo4c'	-0.3506e+03	-0.4279e-01	-0.1862e-04	0.1154e-08	7.18
'ig'	0.2727e+02	-0.4520e-01	-0.2401e-05	0.1800e-09	7.18
'i2g'	0.1806e+02	-0.6586e-01	-0.4376e-05	0.3295e-09	7.18
'hig'	0.8526e+01	-0.5163e-01	-0.3749e-05	0.2327e-09	7.18
'hg'	0.5387e+02	-0.2944e-01	-0.2396e-05	0.1833e-09	7.18
'h2g'	0.2313e+01	-0.3373e-01	-0.3538e-05	0.2409e-09	7.18
'h2og'	-0.5534e+02	-0.4734e-01	-0.4740e-05	0.2648e-09	7.18

Table 7.3 (cont.)

FREE ENERGY DATA FOR THE 99 SPECIES IN VICTORIA

	<u>A</u>	<u>B</u>	<u>C</u>	<u>D</u>	<u>Ref.</u>
'srohg'	-0.4521e+02	-0.6284e-01	-0.6408e-05	0.4624e-09	7.19
'sroh2c'	-0.2190e+03	-0.3231e-01	-0.1818e-04	0.1391e-08	7.19
'sroh2g'	-0.1360e+03	-0.7910e-01	-0.1100e-04	0.7962e-09	7.19
'sroc'	-0.1317e+03	-0.2794e-01	0.1906e-06	-0.8241e-09	7.18
'srog'	-0.2891e+00	-0.5846e-01	-0.3792e-05	0.1433e-09	7.18
'srig'	-0.4105e+01	-0.6864e-01	-0.4403e-05	0.3275e-09	7.19
'sri2g'	-0.6045e+02	-0.8718e-01	-0.7183e-05	0.5493e-09	7.19
'sri2c'	-0.1231e+03	-0.4930e-01	-0.1302e-04	0.9963e-09	7.18
'srg'	0.4194e+02	-0.4173e-01	-0.2399e-05	0.1835e-09	7.18
'src'	0.4951e+01	-0.1757e-01	-0.3613e-05	0.2791e-09	7.18
'sruc4c'	-0.4692e+03	-0.4602e-01	-0.1864e-04	0.1050e-08	7.24
'srzro3c'	-0.4160e+03	-0.3460e-01	-0.1531e-04	0.1054e-08	7.18
'srmoo4c'	-0.3605e+03	-0.3930e-01	-0.1883e-04	0.1167e-08	7.18
'zrog'	0.1432e+02	-0.5297e-01	-0.6391e-05	0.5210e-09	7.18
'zro2g'	-0.6430e+02	-0.6939e-01	-0.6690e-05	0.5112e-09	7.19
'zro2c'	-0.2467e+03	-0.3311e-01	-0.7736e-06	-0.6911e-09	7.18
'zrg'	0.1505e+03	-0.4598e-01	-0.2989e-05	0.1971e-09	7.18
'zrc'	0.1944e+00	-0.7949e-02	-0.5370e-05	0.3999e-09	7.18
'snc'	0.4120e+01	-0.1840e-01	-0.3280e-05	0.2509e-09	7.18
'sng'	0.7591e+02	-0.4391e-01	-0.3045e-05	0.2330e-09	7.18
'snog'	0.7775e+01	-0.5843e-01	-0.4200e-05	0.3083e-09	7.18
'snoc'	-0.6506e+02	-0.1640e-01	-0.6356e-05	0.3524e-09	7.18
'sno2c'	-0.1331e+03	-0.1761e-01	-0.9586e-05	0.6367e-09	7.18
'sno2g'	-7.5870e+01	-5.4570e-02	-6.3560e-06	3.5240e-10	*
'sni2g'	0.3676e+01	-0.8548e-01	-0.7099e-05	0.5350e-09	7.18
'sni2c'	-0.3551e+01	-0.7153e-01	-0.1090e-04	0.8339e-09	7.18
'snteg'	0.4034e+02	-0.6595e-01	-0.4318e-05	0.3305e-09	7.18
'sntec'	-0.1177e+01	-0.3531e-01	-0.8172e-05	0.6757e-09	7.18
'snh4g'	0.4229e+02	-0.5578e-01	-0.1040e-04	0.4490e-09	7.18
'teg'	0.5227e+02	-0.4551e-01	-0.2463e-05	0.1717e-09	7.18
'tec'	0.6646e+01	-0.1965e-01	-0.4341e-05	0.3321e-09	7.18
'te2g'	0.4089e+02	-0.6519e-01	-0.4788e-05	0.3054e-09	7.18
'teog'	-0.2060e+02	-0.6049e-01	-0.4210e-05	0.3090e-09	*
'teo2g'	7.1948e+01	-9.4380e-02	-4.2100e-06	3.0900e-10	7.18
'teo2c'	-0.6775e+02	-0.2324e-01	-0.1325e-04	0.9942e-09	7.18
'te2o2g'	-0.1972e+02	-0.8455e-01	-0.1006e-04	0.7160e-09	7.18
'h2teg'	0.2652e+02	-0.5699e-01	-0.5511e-05	0.3108e-09	7.18
'uo2c'	-0.2534e+03	-0.2414e-01	-0.9969e-05	0.6970e-09	*
'uo2g'	-1.0920e+02	-6.4620e-02	-9.9690e-06	6.9700e-10	7.18
'uc'	0.2852e+01	-0.1251e-01	-0.6735e-05	0.5827e-09	7.18
'ug'	0.1194e+03	-0.4264e-01	-0.5099e-05	0.3901e-09	7.18
'u4o9c'	-0.1052e+04	-0.1053e+00	-0.4490e-04	0.3096e-08	7.18
'u3o8c'	-0.8335e+03	-0.8698e-01	-0.3667e-04	0.2456e-08	7.18
'uo3c'	-0.2869e+03	-0.3053e-01	-0.1185e-04	0.8065e-09	7.18
'krg'	0.0000e+00	0.0000e+00	0.0000e+00	0.0000e+00	N/A
'xeg'	0.0000e+00	0.0000e+00	0.0000e+00	0.0000e+00	N/A
'o2g'	0.2412e+01	-0.5147e-01	-0.3943e-05	0.2630e-09	7.18

Table 7.3 (cont.)					
FREE ENERGY DATA FOR THE 99 SPECIES IN VICTORIA					
	A	B	C	D	Ref.
'og'	0.6138e+02	-0.4064e-01	-0.2404e-05	0.1839e-09	7.18
'fec'	-0.6072e+00	-0.2826e-02	-0.7397e-05	0.6513e-09	7.18
'feg'	0.9952e+02	-0.4411e-01	-0.3120e-05	0.2387e-09	7.18
'feoc'	-0.6730e+02	-0.7124e-02	-0.1286e-04	0.1235e-08	7.18
'feog'	0.6279e+02	-0.6072e-01	-0.4223e-05	0.3089e-09	7.19
'fe2o3c'	-0.1850e+03	-0.3355e-01	-0.1617e-04	0.1170e-08	7.18
'fe3o4c'	-0.2836e+03	0.1782e-02	-0.4917e-04	0.4748e-08	7.18
'fei2c'	-0.7378e+01	-0.5784e-01	-0.1302e-04	0.9963e-09	7.18
'fei2g'	0.2606e+02	-0.8919e-01	-0.7293e-05	0.5282e-09	7.18
'fe2i4g'	0.1304e+02	-0.1425e+00	-0.1545e-04	0.1147e-08	7.18
'fetec'	-0.1316e+01	-0.2360e-01	-0.6575e-05	0.4634e-09	7.18
'fete2c'	-0.1160e+02	-0.2913e-01	-0.1077e-04	0.6349e-09	7.18
'moc'	0.4795e+01	-0.1386e-01	-0.4001e-06	-0.3243e-09	7.18
'mog'	0.1595e+03	-0.4594e-01	-0.2136e-05	0.1312e-09	7.18
'moog'	0.9528e+02	-0.5982e-01	-0.4209e-05	0.3134e-09	7.18
'moo2c'	-0.1364e+03	-0.1485e-01	-0.9593e-05	0.4909e-09	7.18
'moo2g'	0.2642e+00	-0.6884e-01	-0.6486e-05	0.3099e-09	7.18
'moo3c'	-0.1627e+03	-0.2964e-01	-0.1525e-04	0.1204e-08	7.18
'moo3g'	-0.8075e+02	-0.7203e-01	-0.9368e-05	0.7082e-09	7.18
*The data for these gaseous species was obtained by extrapolating the condensed phase data using Tronton's rule [7.25] to estimate the enthalpy of vaporization and a statistical mechanical estimate of the heat capacity.					

Table 7.4	
ELEMENTS TO BE ADDED	
Cadmium	Europium
Indium	Ruthenium
Silver	Cerium
Nickel	
Chromium	

7.5 Radioactive Decay Heating

The treatment of the heating of the reactor vessel environment by decay of radioactive species is described in this section. Expressions are derived for fuel material, aerosol particles, and bulk gases that are contained within the vessel. Decay powers are calculated only for those species that are

identified as "VICTORIA species;" other radioactive species that may be associated with the fuel, but are not tracked by VICTORIA must be treated by some external means.

Volumetric and surface decay powers produce a temperature feedback effect between the fission product species and the vessel environment. Release of radioactive species from the fuel decreases its decay heating, changes its temperature profile, and thus affects the species release via the temperature-dependent chemistry and transport coefficients. Once released from the fuel the same chain of events is again repeated as the radioactive species heat the bulk gas/aerosol flow and structure surfaces. It should be noted that, although VICTORIA does calculate the decay powers, the code does not solve the appropriate energy conservation equations that utilizes these powers to calculate temperature profiles. This solution is performed by other reactor safety codes that VICTORIA will ultimately couple to (e.g., MELPROG or TRAC). Indeed the proper modeling of this temperature feedback would require a "hard" coupling (i.e., information is transferred between codes every time-step) of VICTORIA to one of these thermal hydraulics codes.

7.5.1 Volumetric and Surface Decay Powers

The activity of radioisotope 1 of element i is determined by a decay constant and an isotopic number density:

$$A_{1i} = \lambda_{1i} N_{1i} , \quad (7.63)$$

where A_{1i} has the dimensions of decays per time unit volume. This activity can then be converted into a power density by multiplying by the energy of decay Q_{1i} :

$$P_{1i} = \lambda_{1i} N_{1i} Q_{1i} . \quad (7.64)$$

If we now define γ_{1i} as the fraction isotope 1 present in the number density of element i and a_{ij} as the number of atoms of element i in species j, then the isotopic number density can be written

$$N_{1i} = \gamma_{1i} \sum_{j=1}^J a_{ij} N_j , \quad (7.65)$$

where J is the total number of species under consideration. Substituting Eq.(7.65) into Eq.(7.64), and defining the summation

of P_{li} over elements and isotopes to be the total decay power P gives

$$P = \sum_{i=1}^I \sum_{l=1}^{L_i} \lambda_{li} Q_{li} \gamma_{li} \sum_{j=1}^J a_{ij} N_j, \quad (7.66)$$

where I is the total number of elements under consideration, and L_i is the number of isotopes associated with element i . The dimensions of the total decay power is energy per time per unit volume. Equation (7.66) is used to calculate decay powers for the fuel open porosity region, the bulk gas, and the aerosol particles. For the structure surface decay powers Eq.(7.66) is multiplied by the structure film thickness δ_{st} .

To obtain the radioactive decay data needed for Eq.(7.66) (i.e., λ_{li} , γ_{li} , and Q_{li}), the ORIGEN code [7.2] is used to perform the appropriate nuclear fuel depletion calculation, and thus provide not only the decay data but also the initial radio-isotopic inventory for the fuel geometry release calculation described in Section 7.2. The ORIGEN code is not unique in its ability to perform depletion calculations, other codes exist that can provide this decay and initial condition information to VICTORIA (e.g., the CINDER [7.26] code).

The main assumption behind the use of Eq.(7.66) is the neglect of decay chains. Isotopes are not lost from the number densities when they decay, and the creation of radioactive daughters upon decay of their parents is ignored.

7.5.2 Information for the Code User

The decay powers are calculated in subroutine DECAY for the fuel geometry (which sums both the open porosity and grain contributions), the bulk gas flow (which for this purpose includes the aerosol particles), and for the structure surfaces. The simple calculational flow of subroutine DECAY is illustrated in Fig. 7.8.

7.6 Species Aerosol Behavior and Transport

The aerosol treatment in VICTORIA uses the CHARM model developed by Wheatley [7.3]. CHARM models aerosol behavior in a single computational cell; time-varying external conditions are assumed to have been calculated in advance and are supplied as data to the model. The aerosol particles have a single, constant

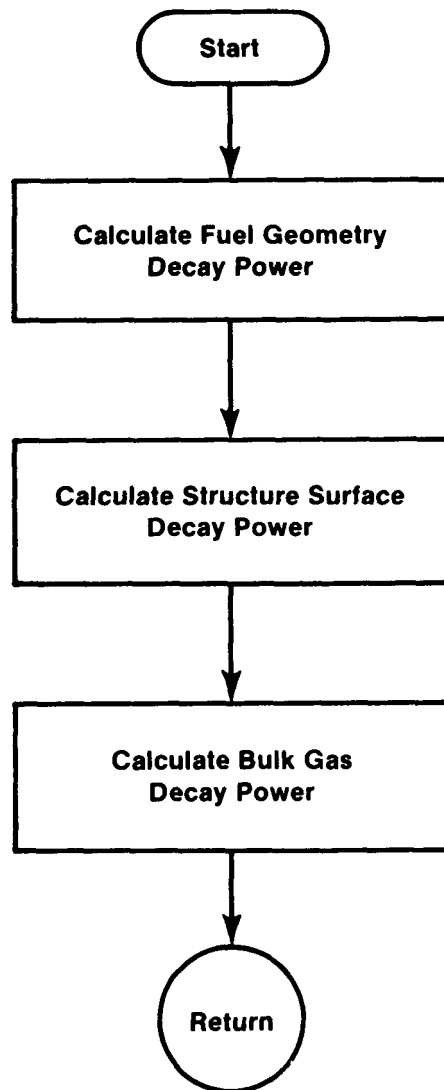


Fig. 7.8 Flowchart of the logic flow in the DECAY subroutine.

composition and can agglomerate, deposit on surfaces and leak from the cell. Transport of aerosol particles between adjacent cells provides a time-varying source and sink of particles for each cell. The agglomeration and deposition models in MAEROS [7.27] have been used in CHARM. However, models have been added to treat deposition by turbulence and to estimate boundary layer thicknesses and turbulence properties of the flow field.

7.6.1 Description of the Model

The governing equation of the aerosol phenomena treated by CHARM is the following

$$\begin{aligned} \frac{\partial C(m,t)}{\partial t} = & \frac{1}{2} \int_0^m K(\mu, m-\mu, t) C(\mu, t) C(m-\mu, t) d\mu \\ & - C(m, t) \int_0^\infty K(\mu, m, t) C(\mu, t) d\mu - R(m, t) C(m, t) + S(m, t) . \quad (7.67) \end{aligned}$$

where $C(m, t)$ is the aerosol number concentration distribution. It is defined so that $C(m, t) dm$ is the number of particles per unit volume with mass in the range m to $m + dm$ at time t . Drake [7.28] gives an excellent survey of this equation.

K is the agglomeration kernel; it is symmetric and takes account of particle-particle collisions due to Brownian motion, differential gravitational settling, and turbulence. The integrand of the first term with K corresponds to production of particles of mass m due to collision and coalescence of particles of masses $m - \mu$ and μ . The integrand of the succeeding term corresponds to destruction of particles of mass m due to collision and coalescence with particles of mass μ . K can depend explicitly on time through changes in the external conditions; gas temperature, pressure and velocity are particularly important.

R is the removal rate for particles of mass m . Leakage, and deposition onto surfaces induced by thermophoresis, diffusio-phoresis, gravitational settling, Brownian diffusion, and turbulence all contribute to R which, like K , may be time-dependent.

S is the number concentration source rate for particles of mass m .

Implicit in Eq.(7.67) are four assumptions. First, the aerosol is well-mixed throughout the cell. This may require the flow in the cell to be turbulent to promote mixing or the cell to be a small part of a larger region within which the aerosol exists and the aerosol has nearly uniform properties within the cell by virtue of its small size. Second, particles are characterized by their mass only, i.e., particles can have a shape which is a function of m but particles of given mass all have the same shape. Clearly, this breaks down when particles having the same mass have different shapes. This affects agglomeration and deposition rates. Such dependence, however, would be extremely difficult to treat computationally. Third, boundary layers and their effect on deposition are not treated in detail. Fourth, particles do not break-up into smaller particles.

Detailed expressions for K , R and S will be considered in the succeeding sub-sections where the primary aim will be to give formulae in detail as they appear in CHARM and to reference their origin. Ranges of validity and possible uncertainties will be indicated where appropriate. The models in CHARM are based on those in MAEROS [7.27] except where extensions have been made to estimate relevant flow parameters and turbulent deposition. S.I. units are used throughout except for molecular weights, which have units kg/kmole.

7.6.1.1 Gas Properties

Agglomeration and deposition rates depend on properties of the gas in the bulk of the cell and adjacent to surfaces, where they are relevant to the calculation of diffusiophoresis.

The gas density. For the bulk gas, the temperature, T , pressure, P , and average molecular weight, W_g , are supplied as input to the model (when a gas has more than one component the average molecular weight is just the mass of one mole of the mixture). The density, ρ_g , is calculated assuming the gas to be ideal as follows:

$$\rho_g = \frac{PW_g}{10^3 RT} \quad (7.68)$$

The factor 10^3 is included because W_g has units kg/kmole.

The dynamic viscosity η_g is estimated from a correlation of data for hydrogen, steam or a mixture of the two. The correlations are as follows

$$\eta_{H_2} = \frac{5 \left[\frac{MkT}{\pi N_o} \right]^{1/2}}{(16 w \sigma^2)} \quad (7.69)$$

where M is the molecular mass of H_2 , k is Boltzmann's constant, N_o is Advogadro's number, w is a Lennard-Jones collision integral and σ is the Leonard-Jones collision diameter for H_2 .

$$\eta_{H_2O} = \eta_r \left(2 \frac{T}{T_r} - 1 \right) \quad (7.70)$$

where η_r and T_r are constants with the values 3.082×10^{-6} and 151.4, respectively.

For a mixture of hydrogen and steam the viscosity is given by

$$\eta_g = \chi_{H_2} \eta_{H_2} / \left[\chi_{H_2} + \chi_{H_2O} \phi_{H_2-H_2O} \right] + \chi_{H_2O} \eta_{H_2O} / \left[\chi_{H_2} \phi_{H_2O-H_2} + \chi_{H_2O} \right] \quad (7.71)$$

where

$$\phi_{kl} = \left[1 + \left(\frac{\eta_k}{\eta_l} \left(\frac{M_l}{M_k} \right)^{1/2} \right)^{1/2} \right]^2 \left(8 \left(1 + \frac{M_k}{M_l} \right) \right)^{-1/2} \quad (7.72)$$

and χ_{H_2} , χ_{H_2O} is the mole fraction of the indicated species in the mixture.

The mean free path l is calculated from standard kinetic theory as follows

$$l = \eta_g \left[\frac{\pi}{2P\rho_g} \right]^{1/2} \quad (7.73)$$

The vapor concentration in the vapor/gas mixture adjacent to a surface, c_s , where the subscript s can stand for c, w or f according to whether the surface is the ceiling, wall or floor (see Section 7.6.1.5 for the definition of these surfaces), is estimated from the ideal gas law as follows

$$c_s = \frac{\chi_s P M_v}{10^3 R T_s}, \quad (7.74)$$

where χ_s is the molar fraction of the vapor in the mixture, M_v is the molecular weight of the vapor and T_s is the temperature of the mixture adjacent to the surface.

7.6.1.2 Flow Properties

Turbulence parameters of the bulk flow are needed for estimating agglomeration and deposition rates. Viscous and diffusion boundary layer thicknesses are needed for estimating Brownian and thermophoretic deposition.

The friction velocity u_* is calculated from formulae based on those given in Schlichting [7.29] for flow through a cylindrical pipe of any aerodynamic roughness and others for flow through aerodynamically smooth pipes of arbitrary cross-sectional shape. By definition, u_* in terms of the Fanning friction factor, f_* , (see below) is given by

$$u_* = U \left[\frac{f_*}{2} \right]^{1/2}, \quad (7.75)$$

where U is the mean flow speed in the cell relative to surfaces in the cell, usually the cell walls.

The Fanning friction factor f_* is implicitly determined from a correlation established by Colebrook [7.29] for flow through cylindrical pipes of any aerodynamic roughness. The correlation is

$$\frac{1}{2\sqrt{f_*}} = 1.74 - 2 \log_{10} \left[\frac{2z_s}{d_h} + \frac{18.7}{Re 2\sqrt{f_*}} \right], \quad (7.76)$$

where z_s is the equivalent sand roughness of the pipe surface adjacent to the flow, d_h is the diameter of the pipe, and Re is the pipe Reynolds number equal to $\rho_g d_h U / \eta_g$.

This correlation reduces to the quadratic resistance formula for fully rough flow ($\rho_g z_s u_* / \eta_g > 70$), derived by von Karman, and Prandtl's universal law of friction for smooth pipes ($\rho_g z_s u_* / \eta_g < 5$).

The equivalent sand roughness z_s equals the height of protrusions on sand roughened pipes of equivalent roughness (this

originates from the pioneering experiments of Nikuradse who measured the pressure drop along pipes roughened with sand of varying grades at Reynolds numbers ranging from somewhat less than 10^3 to greater than 10^6). z_s in general must be determined empirically but values have been established for common materials, which are given in the table below, reproduced from Schlichting [7.29].

z_s for some common materials	
material	z_s (m)
reinforced concrete	.0003 - .003
cast iron	.00026
galvanized steel	.00015
structural and forged steel	.000045
drawn pipes	.0000015

Note that certain types of protrusion, such as regularly spaced ribs perpendicular to the flow, can give rise to values of z_s significantly larger than the height of the protrusion.

The hydraulic diameter d_h is just the diameter of the pipe. However, experiments by a number of authors have shown that the Fanning friction factor of aerodynamically smooth pipes of non-circular cross-section (such as square, rectangular, triangular and annuli) equals that of circular pipes over a wide range of Reynolds number when d_h is generalized as given in Eq.(7.77) below, despite the complications of secondary flows induced in non-circular pipes. d_h is, in general, is given by

$$d_h = \frac{4A}{p} , \quad (7.77)$$

where A is the cross-sectional area and p is the "wetted" perimeter of the pipe perpendicular to the flow. This generalization is used for aerodynamically rough pipes also, even though it has only been verified for smooth pipes.

The critical Reynolds number The above formulae for u_* are valid when the flow is turbulent and fully developed which, for a cylindrical pipe, is the case when $Re \geq 2300$ and well downstream of the pipe inlet. For convenience, u_* is calculated from these formulae when $Re \geq 2300$ irrespective of the cross-sectional shape of the pipe and the downstream distance from the inlet (and other

axial changes). Note that the critical Reynolds number (based on d_h given by Eq.(7.77)) depends, in general, on the pipe shape.

It is assumed that $u_* = 0$ when $Re < 2300$.

The turbulent energy dissipation rate per unit mass ϵ_* is estimated by equating the rate of energy consumption needed to promote steady flow along a pipe of given length to the rate of energy dissipation due to turbulence with the result

$$\epsilon_* = \frac{4U_*^2}{d_h} \quad (7.78)$$

For fully developed turbulent flow in pipes, Laufer [7.31] shows that direct viscous energy dissipation is negligible compared to energy dissipation due to turbulence. Consequently, the above formula is valid and provides an estimate of the turbulent energy dissipation rate averaged of the pipe cross section.

The viscous boundary layer thickness δ_* is estimated from [7.32]

$$\delta_* = \frac{\eta_g}{\rho_g u_*} \quad (7.79)$$

The diffusion boundary thickness δ_D is needed for estimating deposition by Brownian diffusion from a turbulent flow. It is estimated from [7.33]

$$\delta_D = \delta_* / Sc^{1/3} \quad , \quad (7.80)$$

where Sc is the particle Schmidt number given by

$$Sc = \frac{\eta_g}{kTB} \quad (7.81)$$

k is the Boltzmann constant and B is the particle mobility defined in Section 7.6.1.4 below.

7.6.1.3 Initial and Source Distributions

For convenience analytic formulae are used for these although more general formulations could be accommodated. The initial and source number concentration distribution are chosen to be log-normal. Consequently, the initial density distribution, $Y(m,t) = mC(m,t)$, is given by

$$Y(m, t) = \left[\frac{N}{(2\pi)^{1/2} \log_e(\sigma^3)} \right] e^{- \{ \log_e^2 (m/m_g) / 2 \log_e^2 (\sigma^3) \}} \quad (7.82)$$

where the moments N , σ and m_g are defined in Section 7.6.2.2. These moments, however, are not the most convenient to specify values for and so ρ , σ and r_{50} are used in their place as the primary parameters. ρ and σ are defined in Section 7.6.2.2 also. r_{50} is the radius of the spherically equivalent particle of mass m_{50} and density ρ_p .

N and m_g in terms of ρ and r_{50} are

$$m_{50} = \frac{4}{3} \pi \rho_p r_{50}^3, \quad (7.83)$$

$$N = \frac{\rho}{m_{50}} e^{\log_e^2(\sigma^3)/2}, \quad (7.84)$$

and

$$m_g = m_{50} e^{-\log_e^2(\sigma^3)}. \quad (7.85)$$

Similar expressions apply for the source density distribution, $mS(m, t)$, except N and ρ are replaced by dN/dt and $d\rho/dt$ where these are, respectively, the number concentration generation rate (the number generation rate divided by the cell volume) and the density generation rate (the mass generation rate divided by the cell volume) of the source.

7.6.1.4 Particle Mobilities

The particle mobility, B , is estimated from Stoke's mobility law, B_{St} , with a slip correction due to Cunningham [7.34], C_u , as follows

$$B = B_{St} C_u. \quad (7.86)$$

B_{St} is given by

$$B_{St} = \frac{1}{6\pi\chi_d\eta_g r}, \quad (7.87)$$

where the dynamic shape factor, χ_d , is an empirical dimensionless correction factor introduced to account for the aerodynamic effects associated with nonspherical particles, and r is the radius of the spherically equivalent particle of mass m .

$$Cu = 1 + k_a Kn + k_g Kn e^{-k_b/Kn}, \quad (7.88)$$

where Kn is the particle Knudsen number which equals l/r (i.e., the mean free path/the equivalent particle radius) and k_a , k_g and k_b are empirical dimensionless constants. Values for these constants are supplied by the users in the input deck. Typical values used for k_g and k_b (0.4 and 1.1 respectively) are from Davies [7.35]. A value used for k_a is slightly larger than the value obtained by Davies (1.37 c.f. 1.257).

7.6.1.5 Deposition Rates

Five deposition mechanisms are considered which are described in turn.

The surfaces exposed to aerosol are denoted as ceiling, wall, and floor (within the cell and at the cell boundaries) which are horizontal and downward facing, vertical, and horizontal and upward facing, respectively. Surfaces of given orientation are not distinguished in temperature.

The deposition velocity due to gravitational settling v_G is given by

$$v_G = gmB, \quad (7.89)$$

where g is the acceleration due to gravity. Strictly, $(\rho_p - \rho_g)/\rho_p$ should appear as a multiplicative factor in this equation but, keeping to the constraint to duplicate the models in MAEROS, this small error is ignored.

The deposition velocity due to turbulence v_* is estimated from a correlation of data [7.36] for deposition in smooth pipes. The deposition velocity, v_* , and the dimensionless relaxation time, $\tilde{\tau}$, are defined by

$$v_* = \tilde{v}_* u_* \quad (7.90)$$

and

$$\tilde{\tau} = \tau \rho_g u_*^2 / \eta_g, \quad (7.91)$$

where τ is the particle relaxation time given by

$$\tau = \frac{2\rho_p r^2 C_u}{9\eta_g \chi_d} = mB \quad . \quad (7.92)$$

For $\tilde{\tau}$ between $\sim .1$ to ~ 10 the correlation gives

$$\tilde{v}_{*1} = 6 \times 10^{-4} \tilde{\tau}^2 \quad . \quad (7.93)$$

For $\tilde{\tau}$ between ~ 30 to ~ 1000 the correlation gives

$$\tilde{v}_{*2} = .213 \tilde{\tau}^{-1/8} \quad . \quad (7.94)$$

By combining $\tilde{v}_{*1}$ and $\tilde{v}_{*2}$, a good value for $\tilde{v}_*$ across the whole range of $\tilde{\tau}$ is found to be as follows

$$1/\tilde{v}_*^2 = 1/\tilde{v}_{*1}^2 + 1/\tilde{v}_{*2}^2 \quad . \quad (7.95)$$

This correlation is applied in CHARM without restriction on $\tilde{\tau}$.

The deposition velocity due to Brownian diffusion v_B takes account of Brownian diffusion across the laminar sublayer from a turbulent flow. It is given by

$$v_B = .0594 \, kTB/\delta_D = .0594 \, u_* \, Sc^{-2/3} \quad , \quad (7.96)$$

where kTB is the Stokes-Einstein particle diffusivity. The factor 0.0594 is a departure from MAEROS; it is a dimensionless empirical correction factor from Brockmann, et al. [7.37].

Note that this treatment of turbulent deposition, including Brownian deposition from a turbulent flow, does not distinguish aerodynamically rough or smooth surfaces. In fact, the Liu and Agarwal [7.36] experiments were done in aerodynamically smooth pipes. It is clear from the Brownian deposition formula that a laminar sublayer is supposed to exist so this is also applicable to aerodynamically smooth surfaces. However, experimental data [7.38, 7.39] show deposition is increased when surfaces are rough. It may, therefore, be necessary to generalize this treatment at some future time.

The thermophoretic deposition velocity v_{Ts} , where s can stand for c , w or f , is estimated from

$$v_{Ts} = \frac{T - T_s}{T} \frac{9\pi\eta_g^2 r B Br}{\rho_g \delta_*}, \quad (7.97)$$

where Br is a dimensionless factor given by [7.40]

$$Br = b_k \frac{1}{1 + 3b_m Kn} \frac{1}{2 + 1/(\alpha_g/\alpha_p + b_t Kn)}, \quad (7.98)$$

where b_k , b_m and b_t are dimensionless constants that are supplied by the user (1.0, 1.37 and 1.0, respectively are typical values) and α_g and α_p are the thermal conductivities of the gas and particles, respectively.

The diffusiophoretic deposition velocity v_{Ds} , where s can stand for c, w or f, is estimated from

$$v_{Ds} = D_v \frac{dc_s}{dx} \frac{1}{c_s} \frac{f_s}{f_s + (1 - f_s) (W_g/W_v)^{1/2}}, \quad (7.99)$$

where dc_s/dx is the outward facing vapor concentration gradient near the ceiling, wall or floor according to whether s equals c, w or f and f_s is the fraction of the total surface area in a region that is due to wall, ceiling or floor. Comparison with Derjaguin and Yalamov [7.41] shows this estimate to be applicable in the limit $Kn \gg 1$ and $f_s \ll 1$ and when the flow of vapor to or from the surface is purely diffusive.

The net deposition velocity to any surface v_s is estimated by combining the contributions in the following way

$$v_c = v_B + v_* + v_{Tc} + v_{Dc} - v_G, \quad (7.100)$$

$$v_w = v_B + v_* + v_{Tw} + v_{Dw}, \quad (7.101)$$

and

$$v_f = v_B + v_* + v_{Tf} + v_{Df} + v_G. \quad (7.102)$$

Note that since either or both of v_{Ts} and v_{Ds} could be negative (i.e., when T_s is greater than T or dc_s/dx is negative) and v_G is subtracted in Eq.(7.100) some of the v_s may be negative. The equations above are only applied when the result for v_s is positive otherwise v_s is set to zero.

The deposition rate to a surface per unit cell volume, λ_s , is given by

$$\lambda_s = v_s A_s / V , \quad (7.103)$$

where A_s is the area of surface s exposed to aerosol. $R(m,t)$ is then given by

$$R(m,t) = \lambda_c + \lambda_w + \lambda_f . \quad (7.104)$$

All possible deposition mechanisms have not included in this treatment. Examples left out are Brownian diffusion in a laminar flow; deposition mechanisms associated with electrostatic charges; and inertial deposition due to nonlinear mean streamlines (streamlines may be nonlinear due to bends in the flow path or secondary flows induced by turbulence, which can occur in straight flow paths). This last mechanism can be important and would be an obvious next candidate for which to find a suitable treatment.

7.6.1.6 Agglomeration Rates

Four agglomeration mechanisms are considered which are described in turn. For notational convenience the agglomeration rate formulae are given corresponding to the j^{th} and k^{th} collocation points (see Section 7.6.2.1 for their definition).

The particle terminal velocity u_G is the terminal velocity due to gravity of a particle of mass m and is estimated as

$$u_G = gmB . \quad (7.105)$$

Notice this is just v_G but a different symbol is used here to avoid confusion. As for v_G , a factor $(\rho_p - \rho_g)/\rho_p$ should be included but is ignored to keep the equation identical to that used in MAEROS.

The Brownian agglomeration rate ϕ_B is estimated as

$$\phi_B(m_j, m_k) = 4\pi kT (B_j + B_k) \chi_c(r_j + r_k) Fu(m_j, m_k) , \quad (7.106)$$

where B_j and B_k are the particle mobilities at the j^{th} and k^{th} collocation points respectively, χ_c is the collision shape factor which corrects the collision cross-section of particles when they depart from sphericity and Fu is a factor originally introduced by Fuchs [7.42] to correct for particles small in size compared to the molecular mean free path. Fu is given by

$$1/Fu(m_j, m_k) = 1/Fu_1(m_j, m_k) + 1/Fu_2(m_j, m_k) , \quad (7.107)$$

where Fu_1 is

$$Fu_1(m_j, m_k) = \chi_s \frac{r_j + r_k}{kT(B_j + B_k)} \left[\frac{8kT}{\pi} \left(\frac{1}{m_j} + \frac{1}{m_k} \right) \right]^{1/2} \quad (7.108)$$

Fu_2 is modified from unity according to Sitarski and Seinfeld [7.43]

$$Fu_2(m_j, m_k) = 1 + \frac{2 \sqrt{(\tilde{a}_j^2 + \tilde{a}_k^2)}}{r_j + r_k} \quad (7.109)$$

where $\tilde{a}$ is

$$\tilde{a} = \frac{(r+a)^3 - (r^2 + a^2)^{3/2}}{3ra} - r \quad (7.110)$$

and

$$a = B \left[\frac{2kTm}{\pi} \right]^{1/2} \quad (7.111)$$

χ_s in Eq.(7.108) is the particle-particle sticking efficiency which is the probability that particles stick to one another when they collide. Note that it is included as a multiplicative factor in Eq.(7.108) only, as is done in the MAEROS model.

The gravitational agglomeration rate ϕ_G is estimated as

$$\phi_G(m_j, m_k) = \pi \chi_s \chi_{Fu} \chi_c^2 (r_j + r_k)^2 |u_G(m_j) - u_G(m_k)| \quad (7.112)$$

It arises from collisions of particles travelling under the influence of gravity at different terminal velocities. χ_{Fu} is the collision efficiency defined below and χ_s is as defined above.

The collision efficiency. This is a correction factor which is applied to account for the deflection of the particle stream lines from straight-lines when they approach one another. The correction factor, χ_{Fu} [7.43], is given by

$$\chi_{Fu}(m_j, m_k) = \frac{3}{2} \frac{\min(r_j, r_k)^2}{(r_j + r_k)^2} \quad (7.113)$$

where $\min$ stands for the minimum value of r_j and r_k . r_j and r_k are just the radii of the equivalent spherical particles evaluated at the j^{th} and k^{th} collocation point, respectively.

The turbulent shear agglomeration rate ϕ_{*S} is estimated
[7.44] from

$$\phi_{*S}(m_j, m_k) = \chi_s \chi_c^3 (r_j + r_k)^3 \left[\frac{8\rho_g \pi \epsilon_*}{15\eta_g} \right]^{1/2} . \quad (7.114)$$

It accounts for the action of turbulent shear causing particles which follow the instantaneous stream lines to collide with one another.

The turbulent inertia agglomeration rate ϕ_{*I} is estimated
[7.44] from

$$\phi_{*I}(m_j, m_k) = \chi_s \chi_c^2 (r_j + r_k)^2 \left[\frac{512\rho_g \pi^3 \epsilon_*^3}{15\eta_g} \right]^{1/4} |u_G(m_j) - u_G(m_k)| / g . \quad (7.115)$$

It accounts for particles colliding with one another when, due to their inertia, they are unable to follow the instantaneous stream lines.

In principle, a collision efficiency factor should be included in ϕ_{*S} and ϕ_{*I} , analogously to that included in ϕ_G . Dunbar, et al. [7.45] assume this factor is the same as that which appears in ϕ_G (here estimated as χ_{Fu}). However, it is by no means clear that the same factor applies since the flow field near to particles approaching one another due to gravity is not the same as that when particles approach one another due to turbulence. This factor is omitted, as in MAEROS.

The combined agglomeration rate. The contributions defined above are combined as follows

$$K(m_j, m_k, t) = \phi_B(m_j, m_k) + \phi_G(m_j, m_k) + \{\phi_{*S}(m_j, m_k)^2 + \phi_{*I}(m_j, m_k)^2\}^{1/2} \quad (7.116)$$

According to Saffman and Turner [7.44], the turbulence contributions are added quadratically. However, Dunbar, et al. [7.45] point out that the reasoning used by Saffman and Turner also applies to the gravitational contribution which should therefore be added to $K(m_j, m_k, t)$ in the same way. To maintain consistency with MAEROS, the above formulation is used.

Note that all possible agglomeration mechanisms have not been included. Examples left-out are agglomeration in laminar shear

flows and body force effects (e.g., van der Waals and electrostatic forces).

7.6.2 The Numerical Method

7.6.2.1 Discretization of the Governing Equations

Equation (7.67) is discretized with respect to mass using the collocation finite-element method. In this method the governing equation is required to hold at a set of collocation points only (n in total). A finite-element expansion based on values of variables at the collocation points is used. Here, this will be needed for the estimation of the agglomeration integrals.

The particular method used is identical to that studied by Wheatley [7.3] who showed that accurate solutions to the equation could be obtained with a small number of collocation points and the agglomeration kernel evaluated on the n^2 pairs of collocation points only. Consequently, the agglomeration kernel, which is in general time-dependent, can be economically recalculated as the integration of the equation advances in time.

Using a logarithmic scale, m is discretized as follows

$$\log_e(m_i) = \log_e(m_1) + (i-1)h, \quad i=1, \dots, n \quad (7.117)$$

h is a constant which can be found from m_1 , m_n , the smallest and largest values of discretized mass respectively, and n . h is chosen to be constant.

With this choice for m_i it is convenient to choose the mass distribution, given by $Y(m,t) = mC(m,t)$, as the dependent variable in Eq.(7.66), which becomes

$$\begin{aligned} \frac{\partial Y(m,t)}{\partial t} = & \int_0^m K(\mu, m-\mu, t) Y(\mu, t) Y(m-\mu, t) d\log_e(\mu) \\ & - Y(m, t) \int_0^\infty K(\mu, m, t) Y(\mu, t) d\log_e(\mu) - R(m, t) Y(m, t) + mS(m, t) \quad , \end{aligned} \quad (7.118)$$

where $K(\mu, \nu, t) = K(\nu, \mu, t)$ and, for economy of display here and later, the integration limits are shown for μ rather than $\log_e(\mu)$.

When this equation is evaluated at the points m_i , $i = 1, \dots, n$, it is obvious that although the extended trapezoidal rule

could be used to estimate the second integral on the right hand side the first integral will be troublesome. This is where a finite-element expansion is needed. The integrands are expanded as follows

$$K(\mu, \nu, t) Y(\mu, t) Y(\nu, t) \simeq \sum_{j,k} K_{jk} Y_j Y_k g_j(\log_e(\mu)) g_k(\log_e(\nu)) \quad (7.119)$$

where Y_i and K_{ij} are shorthand for $Y(m_i, t)$ and $K(m_i, m_j, t)$ respectively and the i^{th} element g_i is defined in terms of a basic element g (g is also used to denote the acceleration due to gravity but no confusion should arise) by

$$g_i(x) = g((x-x_i)/h) \quad (7.120)$$

and x_i is $\log_e(m_i)$. The precise form of g will be discussed later but note that is always chosen so that $g(0) = 1$ and $g(\pm 1), g(\pm 2), g(\pm 3), \dots = 0$. This guarantees that the expansion in Eq.(7.119) is consistent in the sense that the equation is satisfied identically when μ and ν are located at any of the collocation points.

The particular expansion in Eq.(7.119) has been chosen for three reasons. First, it entails the agglomeration kernel to be evaluated only at the collocation points. Second, since K and Y have not been expanded separately, the summations which result in the discretized equation are over at most two indices which cuts down on computational labor. And third, the same result for the second integral in Eq.(7.118) is obtained as by applying the trapezoidal rule.

Substituting Eq.(7.119) into Eq.(7.118) and evaluating at the i^{th} collocation point gives the following closed set of equations, being the discretized form of the aerosol equation

$$\frac{\partial Y_i}{\partial t} = \sum_{j,k} P_{jk}^i K_{jk} Y_j Y_k - Y_i \sum_j D_j^i K_{ij} Y_j - R_i Y_i + m_i S_i, \quad (7.121)$$

where R_i and S_i are shorthand for $R(m_i, t)$ and $S(m_i, t)$ respectively and $g_j(x_k) = \delta_{jk}$, where δ_{jk} is the Kronecker delta. The terms with P_{jk}^i and D_j^i correspond to the particle production and destruction terms, respectively on the right hand side of Eq.(7.118). The indices in P_{jk}^i and D_j^i run from 1 to n .

$$D_j^i = \int_0^\infty g_j(\log_e(\mu)) d\log_e(\mu) = h \int_{-\infty}^\infty g(x) dx = h \quad , \quad (7.122)$$

as desired, where $g(x)$ must be chosen such that $\int_{-\infty}^\infty g(x) dx = 1$.

$$P_{jk}^i = n_{jk} \int_0^{m_i} g_j(\log_e(\mu)) g_k(\log_e(m_i - \mu)) d\log_e(\mu) \quad . \quad (7.123)$$

This is simplified somewhat by making the transformation $y = \log_e(\mu/m_j)/h$ to obtain

$$P_{jk}^i = h n_{jk} \int_{-\infty}^{\bar{j}} g(y) g\left(\frac{1}{h} \log_e(1 - e^{(y-\bar{j})/h}) + \bar{k}\right) dy \quad , \quad (7.124)$$

where $\bar{j} = i - j$ and $\bar{k} = i - k$. The integral must be calculated numerically and the integration range must be divided-up into sub-ranges since g is generally not smooth. The indices in the coefficient P_{jk}^i in principle take all values from 1, ...n; however, only a small fraction of the coefficients are non-zero. It is straightforward to find the conditions on i , j and k for this to be so. It can be shown that they depend only on $\bar{j}$ and $\bar{k}$. The non-zero values of the coefficients are conveniently stored consecutively using an indexing based on the derived conditions.

The multiplicative factor n_{jk} is introduced as a correction to conserve mass. By integrating Eq.(7.118) with respect to m from 0 to ∞ the following mass balance equation is obtained

$$\frac{\partial \rho}{\partial t} = - \int_0^\infty R(m,t) Y(m,t) dm + \int_0^\infty m S(m,t) dm \quad . \quad (7.125)$$

This equation multiplied by the volume states that the rate of increase of airborne mass equals the rate supplied by sources less the rate removed by deposition and leakage. Clearly, the agglomeration terms have cancelled as one would expect. However, this is not the case for the discretized equivalent to Eq.(7.125) without the factor n_{jk} in P_{jk}^i . In many applications, the removal and source terms in Eq.(7.118) can be small compared to the agglomeration terms. It is therefore important that the agglomeration terms cancel exactly to ensure that the removal and source terms are not swamped by cancellation errors.

The trapezoidal rule is used to estimate ρ (Eq.(7.135) in Section 7.6.2.3 from which the discretized form of the mass balance equation is

$$\frac{\partial \rho}{\partial t} = h \left\{ \sum_{i,j,k} m_i P_{jk}^i K_{jk} Y_j Y_k - \sum_{i,j} m_i D_j^i K_{ij} Y_i Y_j - \sum_i m_i R_i Y_i + \sum_i m_i^2 S_i \right\} . \quad (7.126)$$

The first two terms on the right hand side must cancel exactly, whatever the values of Y_i for $i = 1, \dots, n$. A sufficient condition is therefore to require that the sum of all coefficients of terms involving $Y_r Y_s$ for any r and s must be zero. P_{jk}^i and D_j^i must be related by

$$\sum_i m_i (P_{rs}^i + P_{sr}^i) = m_r D_s^r + m_s D_r^s = h (m_r + m_s) . \quad (7.127)$$

This amounts to $n(n+1)/2$ constraints on n_r only.

Six options for g are provided in CHARM. The element shown by Wheatley [7.3] to give the best results when n is small is

$$\begin{aligned} g(x) &= 1 - |x| , & |x| < 1 \\ g(x) &= 0 , & |x| \geq 1 \end{aligned} \quad (7.128)$$

This choice gives rise to continuous piece-wise linear finite-element expansions. It has the particular advantage here that g is non-negative everywhere and so all the P_{jk}^i are non-negative. This is sufficient to ensure that no component of the solution of the discretized equation changes sign, as must be so on physical grounds.

7.6.2.2 The Solution Method

The ODE's are solved with the Fortran subroutine DEBDF written by Shampine and Watts [7.46]. $n + 5$ coupled ODE's are solved in all; the additional five equations are integrators for the source mass, the leaked mass, and the masses deposited on the wall, floor and ceiling, respectively.

The local absolute error in Y_i is constrained during the integration as follows

$$\delta Y_i < \epsilon / h \text{ minimum}(N, \rho / m_i) , \quad (7.129)$$

where ϵ is a relative tolerance parameter supplied by the user. This test is designed to result in estimates for N and ρ with a relative accuracy equal to or less than ϵ .

7.6.2.3 Moments of the Discretized Distribution

The moments considered are ρ , the aerosol density, N , the aerosol number concentration, m_{50} , the mass median mass, m_g , the geometric mean mass, and σ , the cube root of the geometric standard deviation with respect to mass. These are defined by

$$\rho = \int_0^{\infty} mY(m,t)d\log_e(m) \quad , \quad (7.130)$$

$$N = \int_0^{\infty} Y(m,t)d\log_e(m) \quad , \quad (7.131)$$

$$\rho/2 = \int_0^{m_{50}} mY(m,t)d\log_e(m) \quad , \quad (7.132)$$

$$N \log_e(m_g) = \int_0^{\infty} \log_e(m)Y(m,t)d\log_e(m) \quad , \quad (7.133)$$

and

$$N \log_e^2(\sigma^3) = \int_0^{\infty} \log_e^2(m/m_g)Y(m,t)d\log_e(m) \quad . \quad (7.134)$$

The definitions of m_g and σ are based on the number concentration distribution; the cube power in the equation for σ is conventional.

ρ times the cell volume is the total airborne aerosol mass. Similarly, N times the cell volume is the total number of airborne aerosol particles. m_{50} is sometimes called the mass median particle size. Approximately half the airborne particles have mass less than m_g . σ is a measure of the spread of the distribution. For a lognormal number distribution 68% of the particles have masses in the range m_g/σ^3 to $m_g\sigma^3$ and 68% of the airborne mass derives from particles with masses in the range m_{50}/σ^3 to $m_{50}\sigma^3$. These relationships hold reasonably well for distributions found in practice.

The numerical estimation of these moments is now considered. Their evaluation with the extended trapezoidal rule is straightforward for all except m_{50} and illustrated only for ρ . The estimate for ρ is

$$\rho = h \sum_i m_i Y_i, \quad (7.135)$$

where Y_i is shorthand for $Y(m_i, t)$ and the summation extends over all values for which the indicated index is defined. $Y(m, t)$ has been assumed to decrease to negligible value between m_1 and $m_1 e^{-h}$ and between m_n and $m_n e^h$.

m_{50} generally lies between adjacent grid points so the integral in Eq.(7.132) is estimated by using a finite-element expansion for the integrand. When the linear finite-element is chosen, this is equivalent to using the extended trapezoidal rule to estimate the contribution to the integral up to the grid point immediately below m_{50} and then using linear interpolation between the grid points either side of m_{50} to estimate the remainder.

7.8 References

- 7.1 Reactor Safety Study - An Assessment of Accident Risks in U.S. Commercial Nuclear Power Plants, WASH-1400, NUREG-75/014, U.S. Nuclear Regulatory Commission (October 1975).
- 7.2 M.J. Bell, ORIGEN--The ORNL Isotope Generation and Depletion Code, ORNL-4628, Oak Ridge National Laboratory, Oak Ridge, TN (May 1973).
- 7.3 C.J. Wheatley, CHARM: A Model for Aerosol Behavior in Time-Varying Thermal-Hydraulic Conditions, NUREG/CR-5162, SAND88-0745, Sandia National Laboratories, Albuquerque, NM (August 1988).
- 7.4 D. Anderson, et al., Computational Fluid Mechanics and Heat Transfer, Hemisphere Publishing Corp., New York, NY (1984).
- 7.5 H. Kleykamp, J. Nuc. Mat., 116, p. 86 (1985).
- 7.6 N. Oi, Z.F. Naturforschung, 20a, 1556 (1965). N. Oi, and J. Takagi, Z.F. Naturforschung, and 19a, 1331 (1964). N. Oi and J. Takagi, Z.F. Naturforschung, 20a, 673 (1965).

- 7.7 D. Cubicciotti and J.H. Davies, Nuc. Sci. and Eng., 60, 314, (1976). K. Konash, et al., J. Nucl. Mater., 116, 86 (1983).
- 7.8 R.B. Bird, W.E. Stewart, and E.N. Lightfoot, Transport Phenomena, John Wiley and Sons, Inc. (1960).
- 7.9 P.S. Maiya, J. Nuc. Mat., 40, p. 57 (1971).
- 7.10 M. Silberberg, J.A. Mitchell, R.O. Meyer, and C.P. Ryder, Reassessment of the Technical Bases for Estimating Source Terms, NUREG-0956, U. S. Nuclear Regulatory Commission, Section 6 (1986).
- 7.11 J.L. Collins, et al., Nuc. Tech., 77, 18 (1987).
- 7.12 A.J. Grimley, III and P.J. Maudlin, "Aerosol Formation in Non-Radioactive Species: Implications for the Source Term," Proc. of the International ANS/ENS Topical Meeting on Thermal Reactor Safety, San Diego, CA (February 1986).
- 7.13 T. Lyman, ed., Metals Handbook, V. 1, 8th Edition, Am. Soc. for Metals, Metals Park, OH (1977).
- 7.14 R.M. Elrick, et al., Reaction Between Some Cesium-Iodine Compounds and the Reactor Materials 304 Stainless Steel, Inconel 600 and Silver; Vol.1: Cesium Hydroxide Reactions, NUREG/CR-3197 1 of 3, SAND83-0395, Sandia National Laboratories, Albuquerque, NM (June 1985).
- 7.15 N.N. Yaneko, The Method of Fractional Steps: The Solution of Problems of Mathematical Physics in Several Variables, (M. Holt, ed.) Springer Verlag, New York, NY (1971).
- 7.16 S.R. Brinkley, Jr., J. Chem. Phys., 15, p. 107 (1947).
- 7.17 G. Forsythe and C. B. Moler, Computer Solution of Linear Algebraic Systems, Prentice-Hall, Englewood Cliffs, NJ (1967).
- 7.18 I. Barin, et al., Thermochemical Properties of Inorganic Substances, Springer-Verlag, Berlin (1977).
- 7.19 D.R. Stull, et al., JANAF Thermochemical Tables, 2nd Ed., NSROS-NBS 37 (1971).

- 7.20 P.A.G. O'Hare and H.R. Hoekstra, J. Chem. Thermodynamics, 6, p. 251 (1974). D. W. Osborne, et al., J. Chem. Thermodynamics, 8, p. 361 (1976).
- 7.21 P.A.G. O'Hare, et al., J. Chem. Thermodynamics, 13, p. 1075 (1981).
- 7.22 P.A.G. O'Hare and H.R. Hoekstra, J. Chem. Thermodynamics, 5, p. 851 (1973). R. Kohli and W. Lacom, Thermochimica Acta, 57, p. 155 (1982).
- 7.23 P.A.G. O'Hare, et al., J. Chem. Thermodynamics, 8, p. 845 (1976). P.A.G. O'Hare, et al., J. Chem. Thermodynamics, 12, p. 1003 (1980).
- 7.24 H. Tagawa, et al., J. Inorg. Nucl. Chem, 41, p. 1729 (1979).
- 7.25 H.W. Adamson, A Textbook of Physical Chemistry, Academic Press, New York, NY (1975).
- 7.26 T.R. England, CINDER - A One Point Depletion and Fission Product Program, USAEC Report WAPD-TM-334, Bettis Atomic Power Laboratory and Westinghouse Electric Corp., NTIS, August 1962 (Rev. June 1964).
- 7.27 Gelbard, F., MAEROS User Manual, NUREG/CR-1391, SAND80-0822, Sandia National Laboratories, Albuquerque, New Mexico (December 1982).
- 7.28 Drake, R.L., A General Mathematical Survey of the Coagulation Equation, in Hidy, G.M. and Brock, J.R. (Eds.), "International Reviews in Aerosol Physics and Chemistry," Vol. 3, Pergamon Press (1972).
- 7.29 Schlichting, H., Boundary Layer Theory, McGraw Hill, (1979).
- 7.30 Colebrook, C.F., "Turbulent Flow in Pipes with Particular Reference to the Transition Region Between the Smooth and Rough Pipe Laws," J. Institution Civil Engineers, 12, p. 133 (1939).
- 7.31 Laufer, J., The Structure of Turbulence in Fully Developed Pipe Flow, NACA R 1174, National Advisory Committee for Aeronautics (1954).

- 7.32 Monin, A.S., Yaglom, A.M., Statistical Fluid Mechanics: Mechanics of Turbulence, Vol. 1, MIT Press (1971).
- 7.33 Keller, K., Aerosol Behavior in Closed Containers, Kernforschungszentrum Karlsruhe, KfK 1758 (1973).
- 7.34 Cunningham, E., Proc. Roy. Soc., A83, p. 357 (1910).
- 7.35 Davies, C. N., Proc. Phys. Soc., 57, p. 258 (1945).
- 7.36 Liu, B.Y.H. and Agarwal, J.K., J. Aerosol Science, 5, p. 145 (1974).
- 7.37 Brockmann, J., McMurry, P.H. and Liu, B.Y.H., J. Colloid Interface Sci., 88, p. 522 (1982).
- 7.38 Wells, A.C. and Chamberlain A.C., Br. J. Appl. Phys., 18 p. 1973 (1957).
- 7.39 Chamberlain, A.C., Proc. Roy. Soc., A296, p. 45 (1967).
- 7.40 J.R. Brock, J. Colloid Interface Sci., 17, p. 768 (1962).
- 7.41 Derjaguin, B.V. and Yalamov, Yu I., The Theory of Thermophoresis and Diffusiophoresis of Aerosol Particles and their Experimental Testing, in Hidy, G.M. and Brock, J.R. (Eds.), "International Reviews in Aerosol Physics and Chemistry," Vol. 3, Pergamon Press (1972).
- 7.42 Fuchs, N.A., The Mechanics of Aerosols, Pergamon Press (1964).
- 7.43 Sitarski, M. and Seinfeld, J.H., J. Colloid Interface Sci., 61, p. 261 (1977).
- 7.44 Saffman, P.G. and Turner, J.S., J. Fluid Mech., p. 16 (1956).
- 7.45 Dunbar, I.H., et al., Comparison of Sodium Aerosol Codes, Commission of the European Communities, EUR 9172 en, (1984).
- 7.46 Shampine, L.F. and Watts, H.A., DEPAC - Design of a User Oriented Package of ODE Solvers, SAND79-2374, Sandia National Laboratories, Albuquerque, New Mexico (1979).

8.0 IFCI MODULE

A problem of major concern in the field of reactor safety is the possibility of energetic fuel-coolant interactions occurring during postulated core meltdown accidents. Considerations of this problem are usually couched in the form of a question asking whether or not a possible fuel-coolant interaction (FCI) might be strong enough to breach the reactor containment. Unfortunately, not enough is known about the basic physical processes involved in a FCI to give a definitive answer to this question at the present time.

Past research on FCI phenomena has been both experimental and theoretical in nature, but has not totally succeeded in answering most FCI safety questions. The experiments, for instance, have provided much useful information, but must, because of economic necessity, be much smaller than reactor scale; and FCIs are known to be scale-dependent processes. On the theoretical side, lack of data makes choosing the correct model from among competing models very difficult; without an accurate model of the physical phenomena occurring during an FCI, the experimental results cannot be successfully extended to reactor scale.

A major roadblock to both extending experimental results to reactor scale and to studying the experiments themselves has been the lack of an integrated model capable of realistically simulating all aspects of an FCI. In our view, the desired approach is to develop a tool capable of providing a best estimate of the postulated FCI, based on known physical laws and verifiable models. Because of the current lack of understanding about FCIs, the safest approach (from the viewpoint of confidence in the results) is to carry out the entire calculation using a model based as much as possible on established descriptions of physical phenomena (for instance, the Navier-Stokes equations and accepted heat transfer correlations). The integrated model should ideally also provide the initial conditions for a possible FCI in a reactor meltdown situation.

Because of these needs, we have begun developing an integrated tool to be used in FCI calculations. In the MELPROG code, the model was dubbed the Integrated Fuel-Coolant Interaction (IFCI) module, and is envisioned as being able to calculate the FCI process from start to finish.

This section describes the current state of development of the IFCI module, the key models, notably the dynamic fragmentation model and the surface area transport model, and the use of a stand-alone version of the module to simulate a representative FCI experiment in the Fully Instrumented Test Series, or FITS [8.1]. This experiment series collectively represents the largest body of data on medium-scale FCIs (medium-scale is the closest to full reactor scale presently available).

8.1 THE FCI PROCESS

The FCI process can be roughly divided into four phases: the coarse mixing phase, the trigger phase, the reaction propagation phase, and the hydrodynamic expansion phase. These four phases are useful conceptually, although in reality they may all be occurring simultaneously in different spatial locations of the FCI region.

The coarse mixing phase is characterized by entry of molten material (fuel) into a coolant (water), with accompanying vapor generation, intermixing of the fuel, water, and vapor, and breakup of the fuel into smaller diameter drops (smaller meaning 1-10 cm); this phase occurs on a timescale of 0.1-1.0 s. During this phase, the fuel and water are insulated from one another by a vapor film, which serves to maintain the fuel temperature close to its initial temperature throughout the entire coarse mixing process. Breakup of the molten fuel is thought to be governed by hydrodynamic instabilities, notably the Rayleigh-Taylor and Kelvin-Helmholtz instabilities. These breakup processes are driven by relative velocity differences between the fuel and the other two "fields," water and steam.

The trigger phase occurs when some local disturbance collapses the vapor films around the fuel; this collapse allows direct water-fuel contact, high heat transfer rates to the water, a rapid increase in pressure, and high relative velocities in the vicinity of the trigger. If the triggering event is sufficiently strong, the mixture will enter an explosive propagation phase. The trigger event is not well understood, but is typically observed to occur quickly, on a timescale of around 100 μ s, and is frequently initiated by contact of the fuel with a solid surface.

The explosive propagation phase is characterized by a "reaction zone" which propagates through the mixture region;

within this reaction zone, the coarsely mixed fuel is rapidly fragmented into particles in the 10-100 μm size, with accompanying rapid increase in fuel surface area, release of heat to the water and generation of shock waves. The timescale for reaction zone processes is the same as that for the trigger phase, on the order of 100 μs , and typical propagation speeds are in the 50-300 m/s range. The same hydrodynamic instabilities operative during the coarse mixing phase could also be responsible for the fine fragmentation occurring during the propagation phase, although other mechanisms are also possible (for instance, jet penetration of the fuel by the water [8.2]).

The expansion phase, wherein the steam-water mixture turns its increased thermal energy into work on the surroundings, is more tractable to analysis using existing hydrodynamic codes than the other FCI phases, and consequently has been treated in some detail by various researchers [8.3,8.4]. A recent study by Bohl has included some modeling of the processes occurring during the coarse mixing and propagation phases. The expansion phase generally occurs on a time scale of 10-100 ms.

8.2 PREVIOUS FCI PREDICTIVE MODELS

As stated earlier, a major obstacle to further understanding FCIs and the analysis of experimental data has been the lack of a tool capable of calculating all important aspects of the FCI process. These important aspects include such phenomena as film boiling heat transfer, fragmentation and mixing of the fuel, and shock wave propagation. In the context of reactor safety, a further phenomena of interest is the generation of hydrogen by chemical reaction between the fuel and the steam. Hydrogen generation is important to reactor safety because of the possibility of burning or detonation of the hydrogen posing a threat to the reactor vessel or containment [8.5].

Various models have been used to study isolated aspects of the whole FCI problem, but the results of these efforts have generally been less than satisfactory because of assumptions that simplify or ignore various key phenomena. The most common assumptions are using a constant particle size during the coarse mixing phase [8.6], setting limits on the spatial extent of mixing based on one-dimensional countercurrent flow correlations [8.7], or simply limiting mixing based on "physical intuition." These assumptions tend to limit the extent of coarse mixing, that

is, the amount of fuel and water mixed together; if not much mixture is formed, then a FCI cannot generate enough energy to damage a reactor vessel or containment building, even at maximum theoretical (Hicks-Menzies [8.8]) efficiency. This procedure, unfortunately, does not provide a conservative estimate of the possible effect of an FCI during a reactor accident. Taking the other extreme of total mixing and maximum efficiency does not provide a useful approach either, as this will almost certainly result in a calculated FCI strong enough to breach the containment; such severe conditions are not thought to be likely, but, again, we are not yet in a position to decide how unlikely.

8.3 A PREDICTIVE MODEL FOR FCIS

The current state of knowledge about the physical processes occurring in FCIs, characteristics of existing hydrocodes, and the necessity of calculating FCIs in a reactor safety context were all considerations in the design of the Integrated Fuel-Coolant Interaction module, or IFCI.

Because of the radically different timescales associated with the different phases of a FCI, a desirable feature for an FCI model is to use an implicit numerical hydrodynamics method, in order to exceed the Courant limit [8.9], thereby reducing computation time. The presence of at least three separate material fields in the FCI problem (water, vapor, and molten fuel), all at different temperatures and moving at different velocities, also suggests the use of a multifield method. The multifield approach is also appropriate for the large systems to be analyzed, since these require use of subgrid-scale modeling (flow maps, boiling curves) to allow modeling with a reasonable number of mesh cells. The presence of shock waves during the propagation phase requires compressible hydrodynamics.

The SETS [8.10] method used in FLUIDS is an appropriate hydrodynamic method that satisfies the above requirements for an integrated FCI code. The FLUIDS module is used as the basis for IFCI, with the addition of models for FCI phenomena.

8.4 IFCI IN A MELPROG CONTEXT

The final goal is to use IFCI as a predictive tool for calculation of possible FCIs in a reactor geometry. Some discussion is in order as to how IFCI will interact with MELPROG

in this situation, both as regards to providing the initial conditions for the FCI (calculated by MELPROG) to IFCI, and applying the effects of an FCI (calculated by IFCI), such as material velocities, final melt size, and shock waves, to a MELPROG calculation for determination of material relocation and structural loading.

Several possibilities exist for the MELPROG-IFCI coupling. IFCI is presently running in stand-alone mode; however, the present version of IFCI could easily become part of MELPROG as a module. The simplest approach is to have IFCI use the usual FLUIDS mesh, which is typically very coarse (mesh cell sizes are 50 cm to 1 m in size). Doing the FCI calculation on such a coarse mesh means that the sub-grid-scale modeling (and the correctness of same) is more important in determining the results, than if the calculation were done with a finer mesh. This may or may not be acceptable, depending on the level of confidence placed in the sub-grid-scale models.

The most complete solution (and also the most difficult to implement) would have IFCI rezone the FCI region to a finer mesh. This approach allows use of a fine mesh during the FCI calculation and feedback to the reactor environment during the FCI calculation; however, development of the rezoning algorithm is likely to be nontrivial.

8.5 IFCI CODE DESCRIPTION

The following sections first describe the overall structure and features of IFCI, and then describe the specific models added to MELPROG to implement IFCI.

8.5.1 General Code Structure

The IFCI module consists of the fluid dynamics treatment discussed in Chapter 2 and the following additional models: (1) a dynamic fragmentation model, which calculates the breakup of the melt based on local hydrodynamic conditions (densities and velocities), and (2) an advection model for melt surface area, which handles the transport of surface area in the finite difference mesh when melt in a mesh cell, having some specific surface area, crosses into a neighboring cell containing melt with some other specific surface area.

Other extensions to the FLUIDS treatment necessary to implement IFCI include providing the interfield constitutive

relations between the fourth field (molten fuel or melt) and the other three fields, and extending the equation-of-state package for water-steam to allow higher pressures and temperatures.

8.5.2 Fragmentation Model

The idea of a dynamic fragmentation model, which calculates the characteristic melt diameter as a function of instantaneous hydrodynamic conditions, was first proposed by Camp in Ref. [8.11]. A model using this idea was later incorporated into a version of the TEXAS [8.12] one-dimensional FCI code by Chu [8.13], using an empirical correlation derived from data obtained in the FITS experiments. Another version of a dynamic fragmentation model was developed by Pilch based on correlations to the existing body of drop breakup data [8.14]. The melt fragmentation model in IFCI is adapted from the Pilch model.

The original Pilch model, as implemented in TEXAS, can be expressed as

$$\frac{dD}{dt} = -C_o |v_r| \epsilon^{\frac{1}{2}}, \quad \epsilon = \frac{\rho_f}{\rho_d}, \quad (8.1)$$

where

- D = drop diameter (m),
- t = time (s),
- v_r = relative velocity between the drop and surrounding fluid (m/s)
- ρ_f = fluid density (kg/m³),
- ρ_d = drop density (kg/m³), and
- C_o = a constant ≈ 0.245 .

This formulation was developed as an extension of correlations to the data from liquid-gas droplet breakup experiments [8.14]. The original correlations describe drop breakup in terms of initial conditions (initial relative velocity and droplet diameter), with the results of the correlations being the final drop size and the total time to reach that final size. The correlations were re-derived in terms of instantaneous drop size and velocity to arrive at a dynamic breakup model.

In IFCI, the drop is described by an Eulerian melt field, interacting with the other three fields, also Eulerian. The drop

may either be smaller than a finite difference mesh cell or extend over many of them; melt material in the drop may be advected from one mesh cell to another; and fragmentation takes place only on the drop surface, that is, in cells containing both the melt field and at least one other field. These differences must be considered in going from a Lagrangian description of the drop to one based on an Eulerian description.

The modified Pilch fragmentation model in IFCI is formulated in terms of rate of change of surface area, rather than rate of change of drop diameter; this is because the quantity that is advected with the melt on the Eulerian finite difference mesh is the surface area per unit volume, or volumetric surface area. A surface area formulation also allows treatment of jets and other more general flows, an important advantage for a general predictive tool such as IFCI. The advection algorithm for volumetric surface area will be discussed in detail in the next section. The volumetric surface area A_v and the drop diameter D are related, for spherical drops smaller than a mesh cell, through the melt volume fraction α_m ,

$$A_v = \frac{6\alpha_m}{D} , \quad \alpha_m = \frac{V_m}{V} , \quad (8.2)$$

where

$$\begin{aligned} A_v &= \text{volumetric surface area of the melt (m}^2\text{/m}^3\text{)}, \\ \alpha_m &= \text{melt volume fraction,} \\ V_m &= \text{melt volume in a cell (m}^3\text{), and} \\ V &= \text{cell volume (m}^3\text{).} \end{aligned}$$

Differentiating the expression for volumetric surface area leads to an equation for rate of change of A_v in terms of the rate of change of diameter:

$$\Gamma_A = \frac{dA_v}{dt} = - \frac{6\alpha_m}{D^2} \frac{dD}{dt} = - A_v \frac{1}{D} \frac{dD}{dt} , \quad (8.3)$$

where Γ_A = surface area source (1/m/s).

This expression is used as a surface area source term in the continuity equation for A_v .

The numerical formulation of the surface area source must be done carefully so that the relation between surface area and melt diameter is preserved. In IFCI, a "staggered" mesh cell is used,

where the velocities are defined on the cell edges and densities, volume fractions, drop diameter, the volumetric surface area (and hence the surface area source term), are defined at the cell center (see Figure 8.1).

Another numerical consideration peculiar to the SETS method as implemented in MELPROG is that there is a minimum volume fraction in a cell for each field, even if the field is not actually present; this minimum volume fraction must be taken into account when forming averages on cell boundaries of field densities and volume fractions for use in the breakup model, so that the actual property values are not affected by residual values in the empty cells.

The fragmentation model is set up in IFCI by first calculating the rate of change of diameter on each of a cell's edges. The rate is calculated only if both the melt field and at least one other field are present in the two cells adjoining a given edge. The cell-centered quantities are averaged so that the averaged quantity will go to the correct limit under bounding conditions, for instance one cell full of melt and the adjoining cell full of water.

The following description of the finite difference formulation for the averages is written in terms of averages on the top cell edge; averages in the radial direction are done in a completely analogous fashion.

The volume fractions on the edge are formed as the simple arithmetic average,

$$\bar{\alpha}_1 = \frac{\alpha_{1ij} + \alpha_{1ij+1}}{2}, \quad \bar{\alpha}_2 = \frac{\alpha_{2ij} + \alpha_{2ij+1}}{2}, \quad (8.4)$$

where

- $\bar{\alpha}_1$ = average volume fraction for field 1 (vapor) on the top edge of cell (i,j),
- $\bar{\alpha}_2$ = average volume fraction for field 2 (water) on the top edge of cell (i,j),
- i = radial cell index,
- j = axial cell index,
- α_{kij} = cell volume fraction for field k in cell (i,j).

The water and vapor field densities are first averaged separately by weighting with the cell volume fractions in the two cells to give effective water and vapor densities on the cell edge,

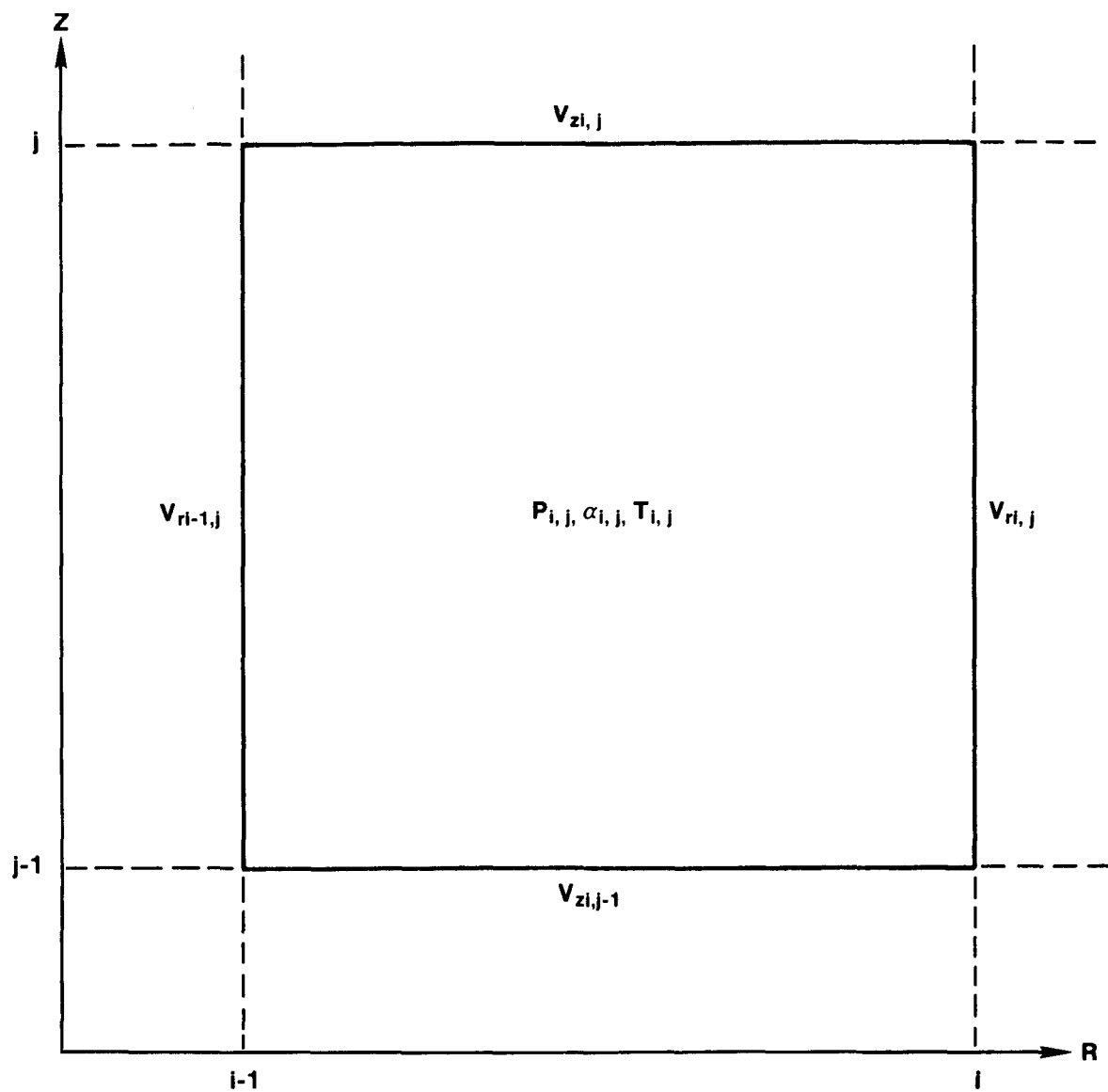


Fig. 8.1 Finite difference mesh cell and variable location in IFCI.

$$\bar{\rho}_k = \frac{\alpha_{kij}\rho_{kij} + \alpha_{kij+1}\rho_{kij+1}}{\alpha_{kij} + \alpha_{kij+1}} . \quad (8.5)$$

The effective fluid density for use in the breakup correlation is then formed by weighting the effective edge densities with the respective edge volume fractions,

$$\rho_f = \frac{\bar{\alpha}_1\bar{\rho}_1 + \bar{\alpha}_2\bar{\rho}_2}{\bar{\alpha}_1 + \bar{\alpha}_2} . \quad (8.6)$$

This procedure is used so that if, for instance, one cell is full of water and the other contains melt, the fluid density calculated will be equal to the water density, rather than one-half the water density. The effective fluid velocity is calculated as the effective-mass-weighted velocity normal to the edge. For the example case, the axial velocity is the one used. This choice is the correct one for the present Rayleigh-Taylor model, which is driven by accelerations normal to the interface between the melt and fluid, whereas other instabilities, such as Kelvin-Helmholtz, would be driven by tangential velocities. The fluid velocity is given as

$$v_f = \frac{\bar{\alpha}_1\bar{\rho}_1 v_{z1} + \bar{\alpha}_2\bar{\rho}_2 v_{z2}}{\bar{\alpha}_1\bar{\rho}_1 + \bar{\alpha}_2\bar{\rho}_2} , \quad (8.7)$$

where

v_f = fluid velocity (m/s) and
 v_{zk} = axial velocity for field k on top edge of cell (m/s).

The effective melt diameter used in the rate model on the cell edge is formulated as

$$\bar{D} = (\alpha_{mij} + \alpha_{mij+1}) \left[\frac{\alpha_{mij}}{D_{ij}} + \frac{\alpha_{mij+1}}{D_{ij+1}} \right]^{-1} . \quad (8.8)$$

This formulation is consistent with the transported quantity, volumetric surface area, which is proportional to $1/D$, and also allows handling of the minimum volume fraction cutoff used in the IFCI numerics. The minimum volume fraction is typically 10^{-5} ; cells that are turned off are set to a minimum ten times smaller. This means that cells also have a minimum volumetric surface area, whether there is actually any melt in them or not. We

would like this residual surface area to be small in comparison to the amount of surface area fluxed into an empty cell during one timestep. If these conditions are not met, then the incoming surface area can be swamped by the residual amount in the receiving cell. In IFCI, the initial diameter in empty cells is set to a large number, 10^5 m. This reduces the residual area to around 10^{-10} .

Setting D in empty cells to a large number and using a geometric averaging procedure causes the geometric average of adjoining empty and full cells to be equal to the melt diameter of the full cell, which is the desired result.

The quantity calculated on the cell edges is actually the rate of change of diameter divided by the effective diameter; this quantity is weighted by the effective melt volume fraction for each edge and used to form an effective rate of change for the cell,

$$\Gamma_A = - A_v \sum_{l=1}^4 \bar{\alpha}_l \left(\frac{1}{D} \frac{dD}{dt} \right)_l \bigg/ \sum_{l=1}^4 \bar{\alpha}_l , \quad (8.9)$$

where the sum on l goes over the four cell edges.

The above expression for rate of change of volumetric surface area in a cell due to Rayleigh-Taylor instabilities is then used as a source in the surface area transport equation. After calculation of the new surface area using the transport equation, the new melt diameter is obtained using a relation between diameter, volume fraction, and surface area.

If the melt extends over more than one cell, the formulation of the fragmentation model must be modified, because fragmentation is assumed to occur only in cells containing the interface between the melt and the other fields, and the original formulation is for global breakup of one drop. The global model, derived for the entire mass of melt and for the fluid conditions surrounding the melt, must be expressed in terms of local cell conditions. One way to accomplish this is to note that the total rate of change of diameter, given by the Pilch model, is equivalent to the volumetric surface area source multiplied by the volume of the melt mass, or total rate of change in surface area. The same total rate of change in surface area must then be generated by those cells on the melt surface; this will be true if the surface source in those cells is increased by a factor f_s .

relating the total melt volume to the melt volume in the surface cells. The factor f_s has somewhat different values depending on the assumed melt shape and what geometry the finite difference cells have. Assuming a spherical melt drop, uniform-sized cells with cell length Δx , Cartesian x-y-z geometry, and $\Delta x \ll D$, the ratio is equal to

$$f_s \approx \frac{D}{6 \Delta x} . \quad (8.10)$$

In practice, the value of D available is a local value that may vary from cell to cell. The cell dimensions are also not necessarily uniform, and the drop may not be spherical. The procedure used in IFCI is to divide the rate of change of diameter calculated on the cell edges by the length of the cell edge, rather than by $\bar{D}$. This gives a correction that differs from the above equation for f_s by a factor of 6, but goes to the limit for $D \leq \Delta x$ as $D \rightarrow \Delta x$.

8.5.3 Surface Area Transport

A consideration arises in an code using an Eulerian finite difference mesh that is not present in one using a Lagrangian description, in that the melt material and its properties must be advected and conserved as the melt moves across mesh cell boundaries. In particular, the surface area per unit volume is advected and conserved in IFCI. The surface area transport equation is formulated as

$$\frac{\partial A_v}{\partial t} + \nabla \cdot \mathbf{v}_m A_v = \Gamma_A , \quad (8.11)$$

where $\mathbf{v}_m$ is the melt velocity vector. The surface area transport equation is solved in IFCI after the mass and energy stabilizer step in the SETS method [8.10]. The new values of the characteristic melt diameter are then calculated from the new volumetric surface areas as

$$D = \frac{6 \alpha_m}{A_v} . \quad (8.12)$$

8.6 FCI EXAMPLE PROBLEM

The initial test problem for the IFCI code has been used both to demonstrate the capabilities of IFCI and to detect problems in

the numerics and/or phenomenological models. The test problem is a representative FITS experiment, using typical experiment conditions as the input to IFCI.

The following sections briefly describe the problem setup, the results of the coarse mixing phase, and the results of the triggering/propagation phase; calculation of the expansion phase was not carried out completely, both because the expansion phase has already been studied in detail by previous researchers [8.4], and because this phase does not present any special calculational difficulty for IFCI. The main purpose of the test calculation was to demonstrate the operation of the IFCI fragmentation models during the coarse mixing and propagation phases.

8.6.1 Problem Setup

The problem conditions are patterned after the FITS "D" series experiments [8.1]. In this series, roughly 20 kg of molten iron-alumina thermite was dropped into water 51-66 cm deep contained in a square lucite chamber 61 cm on a side. The velocity of the melt on entry into the water was 5.7-7.3 m/s. Water temperature ranged from saturation temperature (368 K) at 0.085 MPa to 284 K.

The problem was set up in IFCI's r-z geometry using 5 radial nodes and 10 axial nodes (see Fig. 8.2); although this mesh is fairly coarse, it is adequate for testing the code. The water chamber is approximated as a cylinder 61 cm high by 69 cm in diameter; the cross-sectional area is thus the same as the square lucite chamber in the experiments. The initial water temperature was 373 K, or saturation temperature at 0.1 MPa pressure. The initial melt temperature was 2700 K, and an entrance velocity of 5.9 m/s was used. The initial melt configuration was approximated as a cylinder 20 cm in diameter and 21 cm long, with a total melt mass of 25 kg. These conditions are similar to those in the FITSOD test, except that the test problem has a somewhat larger total melt mass and water depth.

Figure 8.2 shows the initial position of the melt and water on the problem grid. As shown, the tail of the melt enters the problem grid through an inlet boundary on the top of the inner two radial nodes. The outlet boundary is a pressure boundary on the outside of the top axial node. All other boundaries are fixed, which means that pressures generated during the

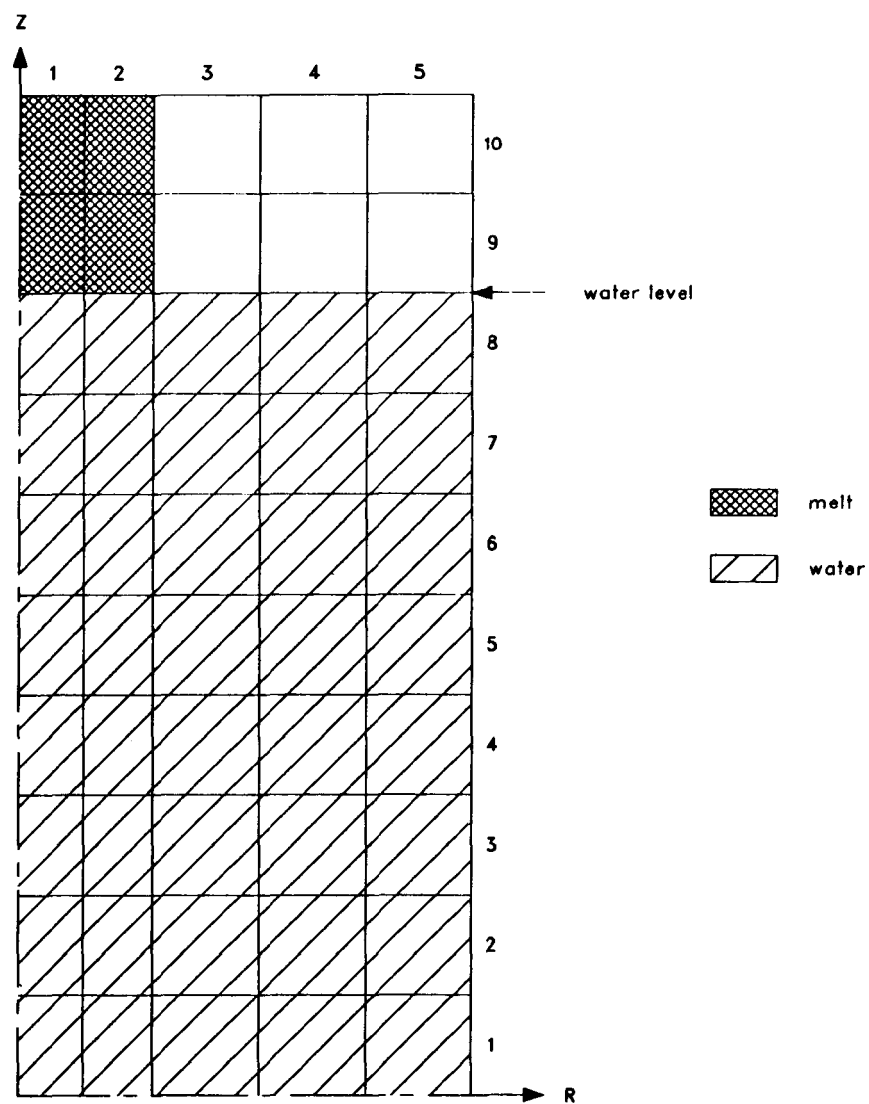


Fig. 8.2 IFCI test problem mesh setup.

propagation phase of the FCI will not be relieved in the same manner as if the chamber was lucite (essentially a free boundary). The calculation is started at the time of initial melt-water contact.

8.6.2 Coarse Mixing Phase

In the coarse mixing phase, the melt enters the water and falls to the bottom of the water chamber. Heat transfer between the hot melt and the surrounding water is limited by film boiling. The melt fragments and mixes with the surrounding water-steam mixture, spreading out radially as it falls. Experimentally, what is observed is a roughly paraboloid shape for the coarse-mixing region (Fig. 8.3). This shape forms fairly quickly after melt-water contact, and persists as the melt mass falls through the water. The experimental observation is in contrast to a theoretical coarse mixing correlation by Corradini [8.15] that predicts an exponential spreading of the mixture region with increasing fall depth.

Two-dimensional slices through the problem, produced by the TRAP graphics postprocessor, are shown at several times during the fall phase in Figs. 8.4-8.6. These figures show the melt volume fraction distribution in the problem at various times. It can be seen that there is an immediate spreading in the radial direction as the melt enters the water surface (Fig. 8.4). The melt then rapidly attains a paraboloid shape as it falls through the water (Fig. 8.5-8.6). This is in agreement with the experimental observations. An extremely interesting feature is the appearance of a steam chimney in the interior of the paraboloid mixture region; the presence of this chimney has been postulated, but never experimentally observed, due to problems with instrumentation in the hostile FCI environment.

This calculation is the first time the chimney's presence has ever been verified, if only by a numerical simulation. The steam volume fraction in the chimney at the time of melt contact with the bottom (0.3 s) is around 0.25.

The water/melt mass ratio in the mixture region is thought to be an important parameter for characterizing FCIs, as the maximum theoretical efficiency, for instance, goes through a peak when the water/melt ratio is varied. The peak occurs at a ratio of 0.5 for thermite/water mixtures at atmospheric pressure. The ratio in the present calculation, determined as the mass-weighted

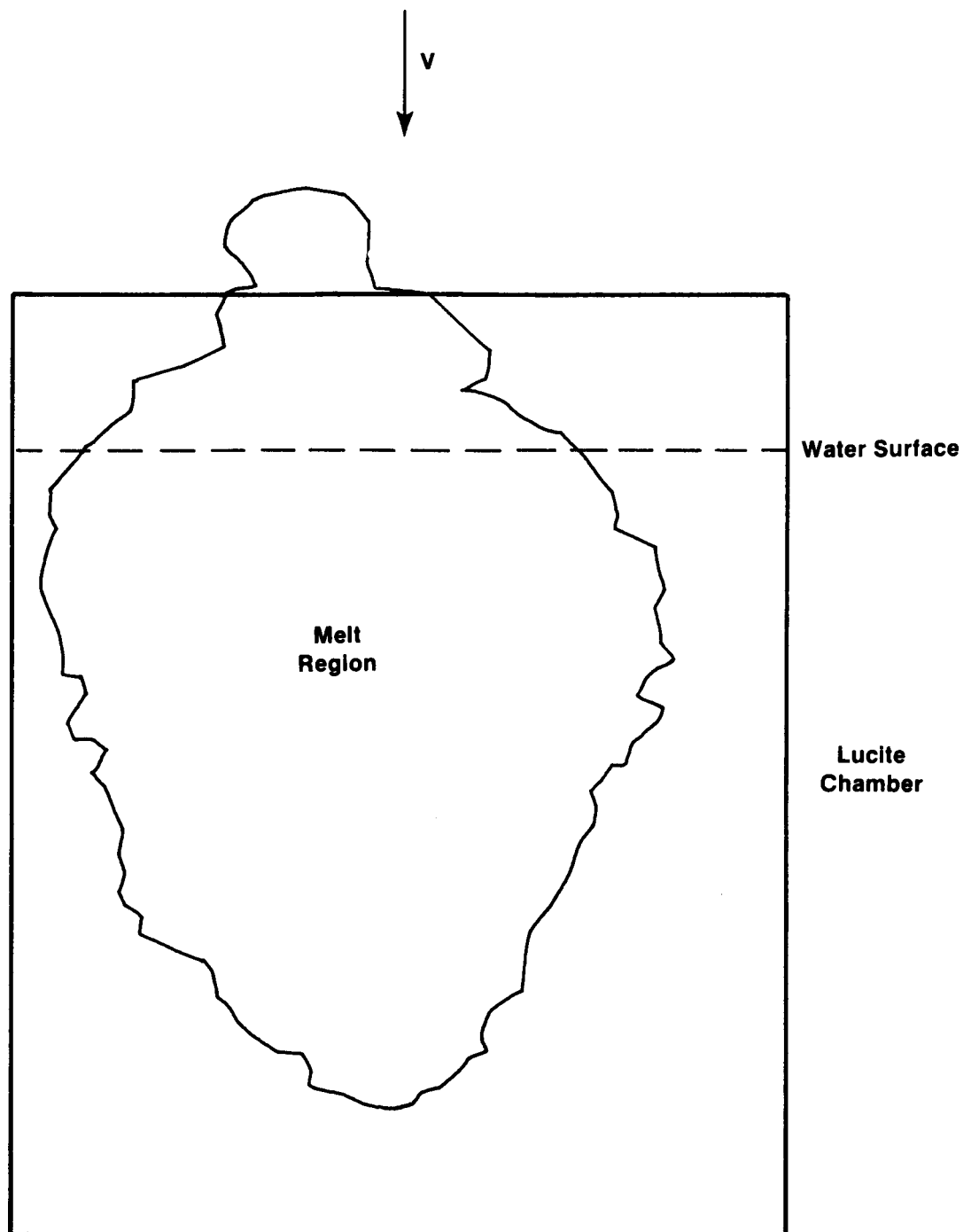


Fig. 8.3 Melt outline in FITS experiment.

FITSOD 2D, 10 AXIAL, 5 RADIAL MODEL

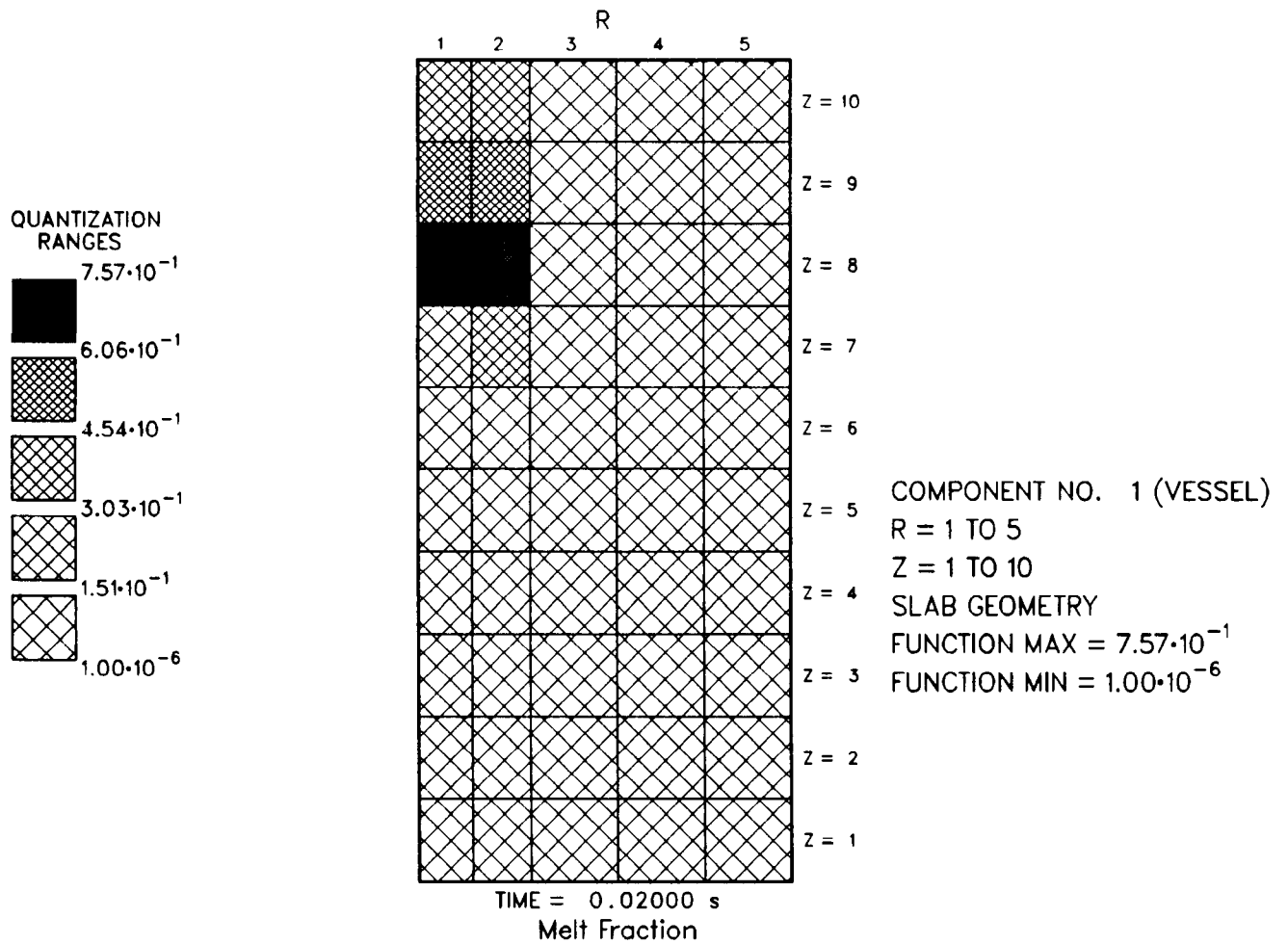


Fig. 8.4 Melt volume fraction at time 0.02 s.

FITSOD 2D, 10 AXIAL, 5 RADIAL MODEL

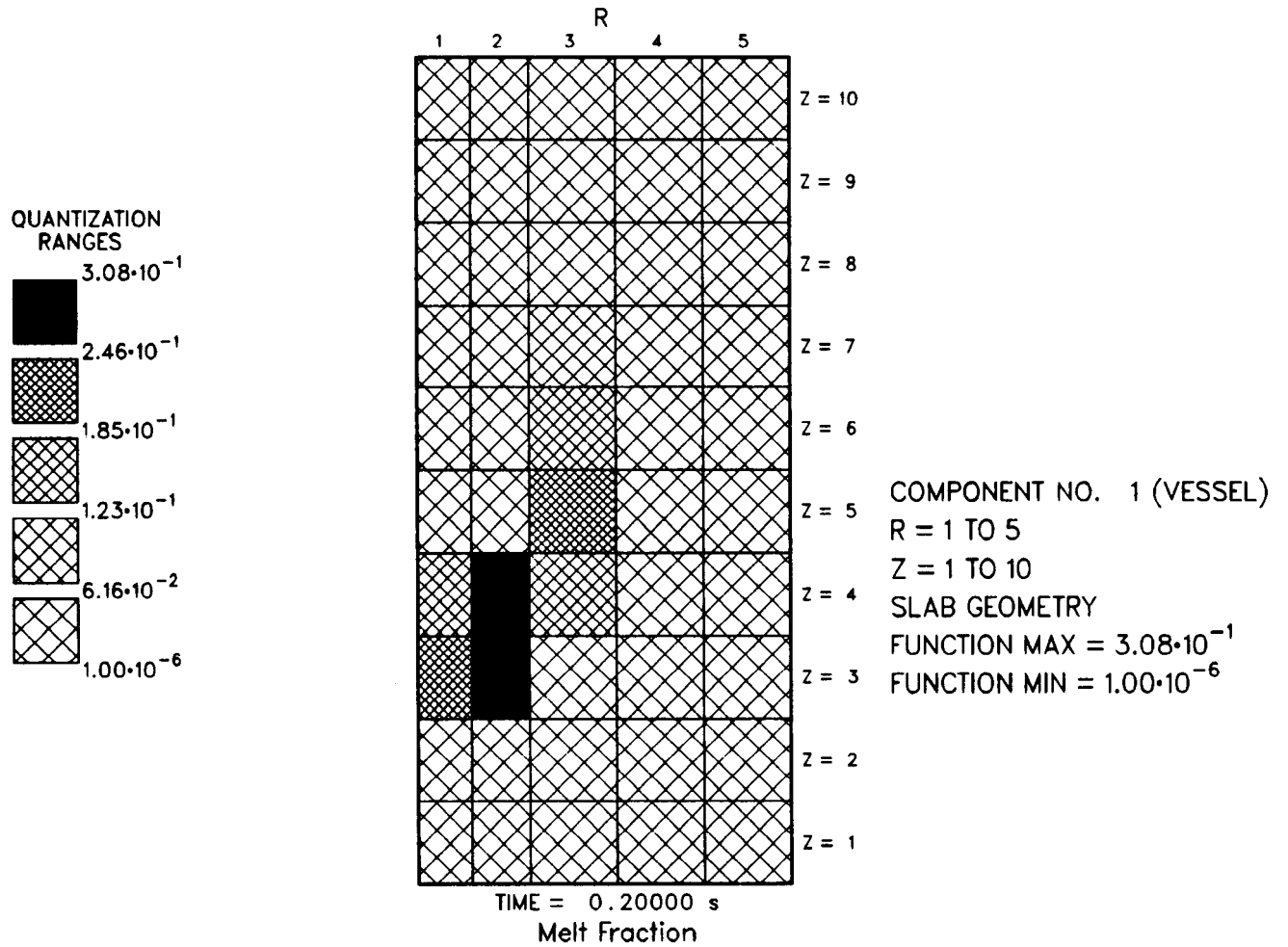


Fig. 8.5 Melt volume fraction at time 0.2 s.

FITSOD 2D, 10 AXIAL, 5 RADIAL MODEL

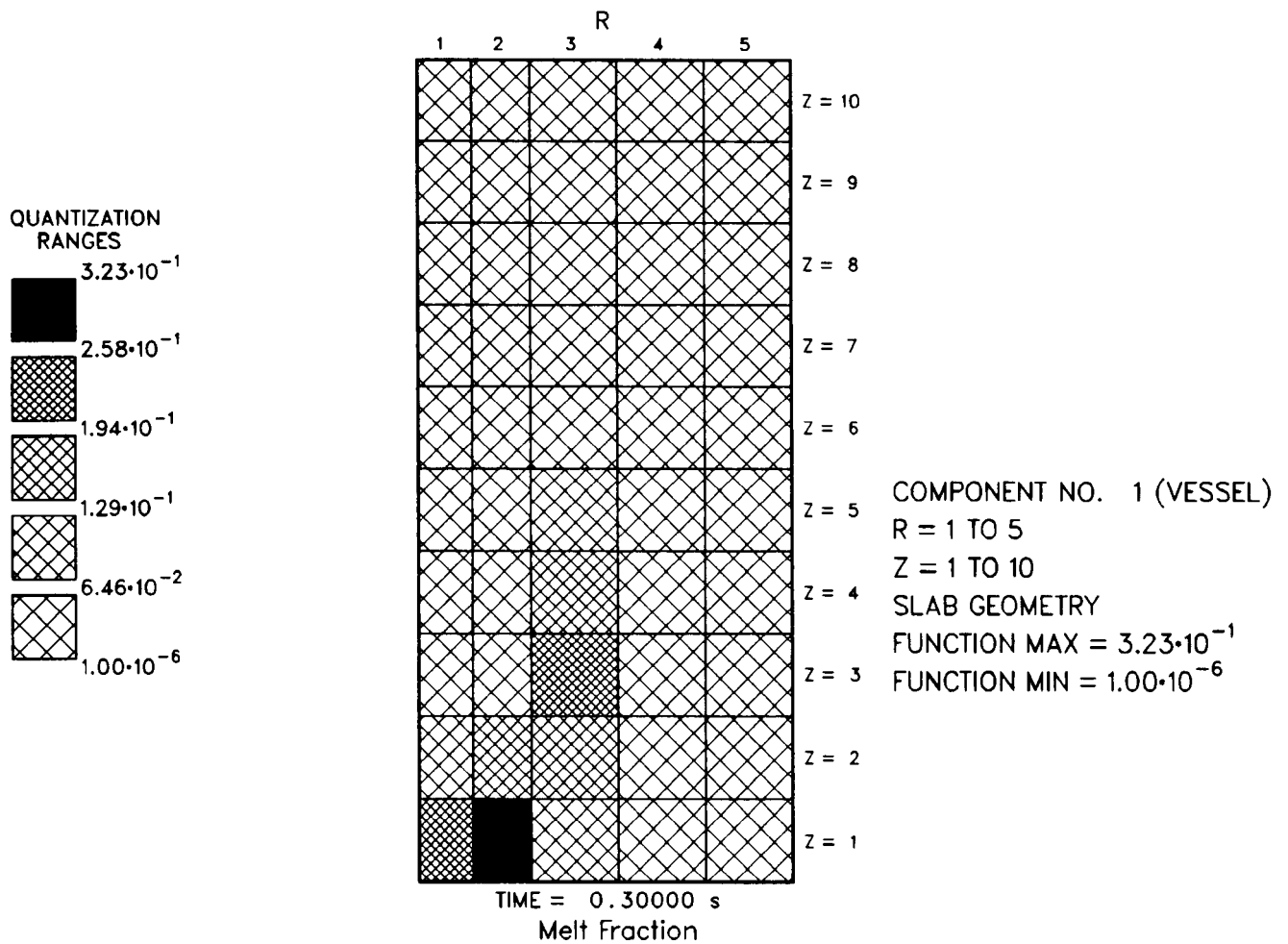


Fig. 8.6 Melt volume fraction at time 0.3 s.

mean of those cells containing melt, is 3.0. A calculation for this case done using experimental procedures would give a ratio of 8.2. (The usual procedure is to divide the total water mass in the tank by the total melt mass.)

The results of the previous paragraphs demonstrate the valuable guidance that an integrated code such as IFCI can provide in interpreting experiments. The code results provide an "x-ray" view of the experiment that has proven impossible to obtain solely by instrumentation.

The melt characteristic diameter decreased during the fall from an initial value of 20 cm to a value at 0.3 s of around 15 cm. This is in agreement with TEXAS predictions of melt breakup using the same initial conditions and the Pilch breakup model. This final mixing diameter is quite large; a diameter of around 1-2 cm is usually cited as the optimum coarse mixing size to produce strong FCIs during the propagation phase.

8.7 TRIGGERING AND PROPAGATION PHASE

To simulate the triggering and propagation phase, the mixture was artificially triggered at 0.3 s, roughly when the melt region contacted the bottom of the water chamber. The initial trigger generated high pressures and velocities, which led to a propagating pressure wave through the mixture.

8.7.1 Triggering

The FCI was triggered by introducing a large surface area source in the bottom center node of the problem. The generation rate was such as to decrease the characteristic diameter of the melt in this bottom center node from 10 cm to 100 μm in a time period of 500 μs . The final melt diameter was chosen as typical of observed debris sizes after a FCI. The time period over which fragmentation occurs is based on observed propagation speeds and reaction zone lengths from FITS tests, but is strictly a ballpark number.

8.7.2 Propagation

The pressure pulse resulting from the final trigger parameters reaches a maximum of about 10 MPa in the trigger cell and nearest neighbors. This level of pressure induced sufficient relative velocity (10-20 m/s) to cause very fine fragmentation,

resulting in final sizes around $100\text{ }\mu\text{m}$ along the center axis. The pressure and relative velocity die away fairly rapidly with increasing radius, resulting in a distribution of melt diameters that is smallest on the axis and becomes larger with radius. Figure 8.7 shows the radial distribution of melt diameter at a time of 0.31 s (roughly the end of the propagation phase) and an axial level of 19 cm (roughly one-half the axial extent of the mixing zone). Although the diameter is around $100\text{ }\mu\text{m}$ on the center axis, the diameters further away from the axis, in the cells containing the bulk of the melt, are in the 1-4 mm range; this diameter is too large to transfer significant heat from the melt to the water over the timescale of the propagation.

The time for the propagation phase to complete was roughly 10 ms (0.31 s). The speed of propagation varied with local three-phase mixture conditions, ranging from 270 m/s close to the trigger cell (a region containing fairly high volume fractions of melt and liquid water) down to 60 m/s at the top of the steam chimney (a region containing mostly vapor).

Figure 8.8 is a three-dimensional plot of pressure versus r and z at 0.304 s which shows the advancing pressure front. Note that the level pressure region behind the pressure front is around 2.5 MPa, about one-quarter the magnitude of the initial trigger pulse.

The total internal energy of the melt at 0.3 and 0.31 s was used to determine the energy given up to the water during the propagation phase. From 0.3 s to 0.31 s, a period of 10 ms, the energy given up was 2.77 MJ, corresponding to a drop in average melt temperature of 104 K; this amount of energy transfer is not large compared to the theoretical maximum and indicates that a large portion of the melt did not participate in the FCI. These small changes in average melt temperature reflect the large final diameter (1-4 mm) for the bulk of the melt. The temperature drop in the melt regions along the center axis, where the final melt diameter was around $100\text{ }\mu\text{m}$, was around 850 K.

An estimation of the efficiency of the FCI can be made by assuming that the 850 K drop in melt temperature corresponds to the theoretical maximum efficiency of 30%; dividing the average temperature drop by 850 K and multiplying by the maximum efficiency then gives an actual efficiency of around 4%.

The melt size distribution was derived at 0.31 s, the end of the propagation phase, by binning the melt mass and diameter in

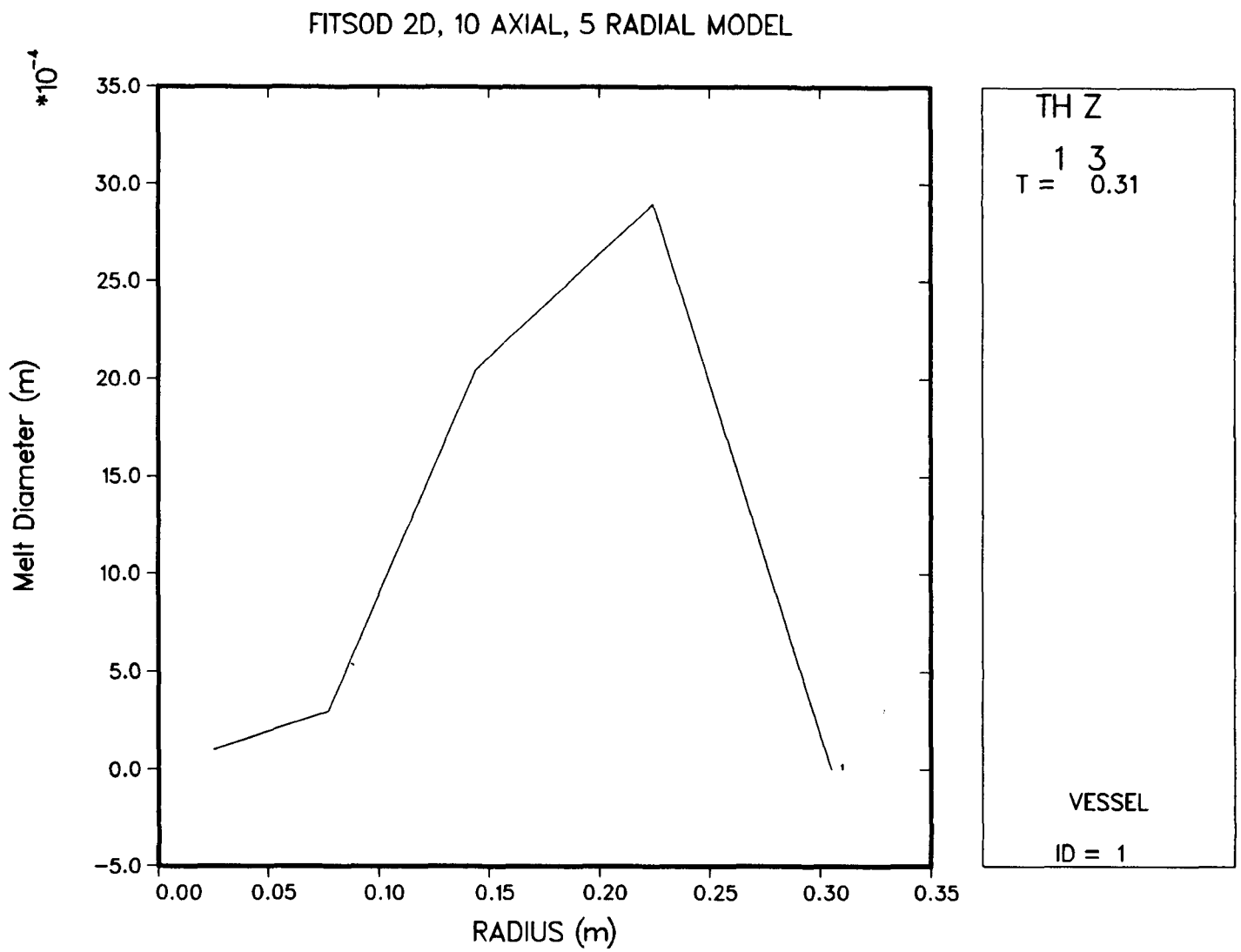
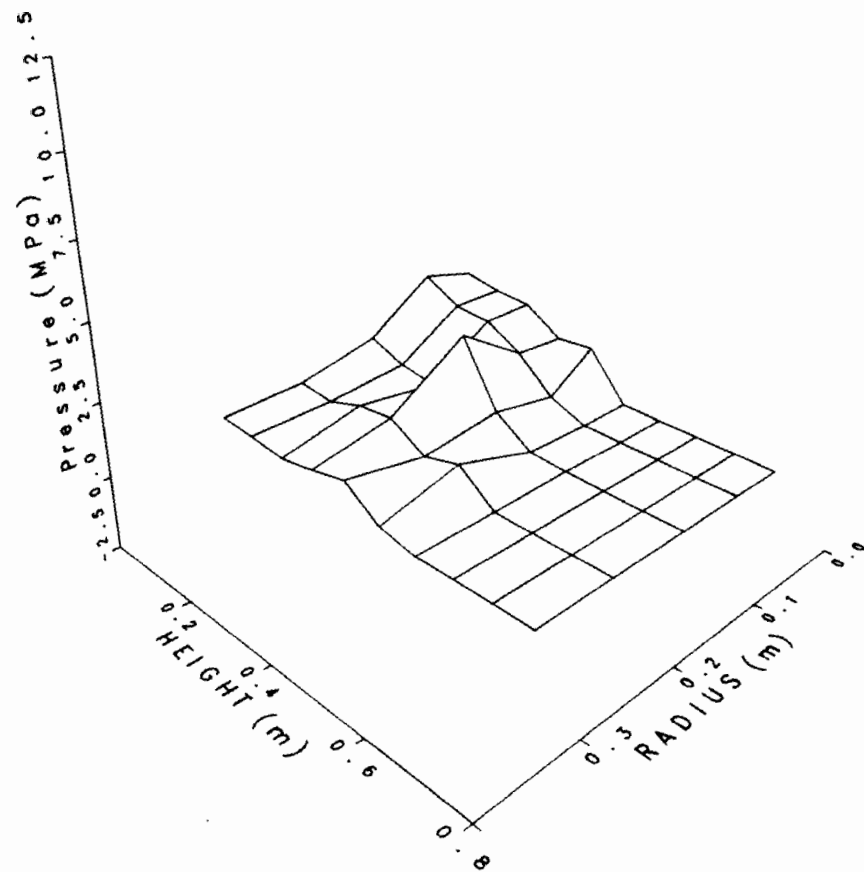


Fig. 8.7 Melt diameter versus radius and height at time 0.31 s.



VESSEL ID = 1
TIME= 0.304 THETA LEV= 1

Fig. 8.8 Pressure versus radius and height at time 0.304 s.

each cell into size ranges corresponding to standard sieve sizes used for sieving debris, normalizing with the total melt mass, and dividing by the size range for each bin. The result is shown in Fig. 8.9, along with the debris distribution from the FITSOD test, an experiment which did not explode, but did have an "eruption," or mild explosion. Since the IFCI test problem does not seem to represent a strong FCI, this comparison is probably the most reasonable.

The distributions match fairly well over the entire range of debris sizes, although some of the smaller sizes are absent from the IFCI results. The absence of the very small sizes is very likely due to the lack of a Kelvin-Helmholtz stripping model in IFCI, which would produce very fine fragments. There is also a peak in the FITSOD data in the largest bin which is not present in the IFCI test. The generally good agreement between experiment and IFCI is an indication that the same mechanisms that produced the experimental distribution are simulated in IFCI with the Pilch breakup model.

Also derived were the various length scales used to characterize debris, which include the Sauter mean diameter, mass median, length mean, surface mean, and volume mean. The two most common length scales are probably the Sauter mean diameter, which is the diameter of a particle which has the average volume per unit surface area, and the mass median diameter, which is simply the diameter for which 50% of the mass has an equal or smaller diameter. These two length scales from the IFCI results are shown in Table 8.1, along with results from the FITS "D" series.

Table 8.1			
Length Scales from IFCI and FITS "D" Series			
TEST	EVENT	SAUTER MEAN DIAMETER (mm)	MASS MEDIAN DIAMETER (mm)
IFCI	SE	1.26	4.0
FITSOD	NE(ER)	1.5	4.6
FITS2D	NE(BC)	3.1	16.0
FITS5D	SE	0.25	0.48
FITS8D	NE(ER)	0.81	3.4
NE - no explosion SE - steam explosion BC - bottom contact ER - eruption			

FITSOD/IFCI D at 0.31 s

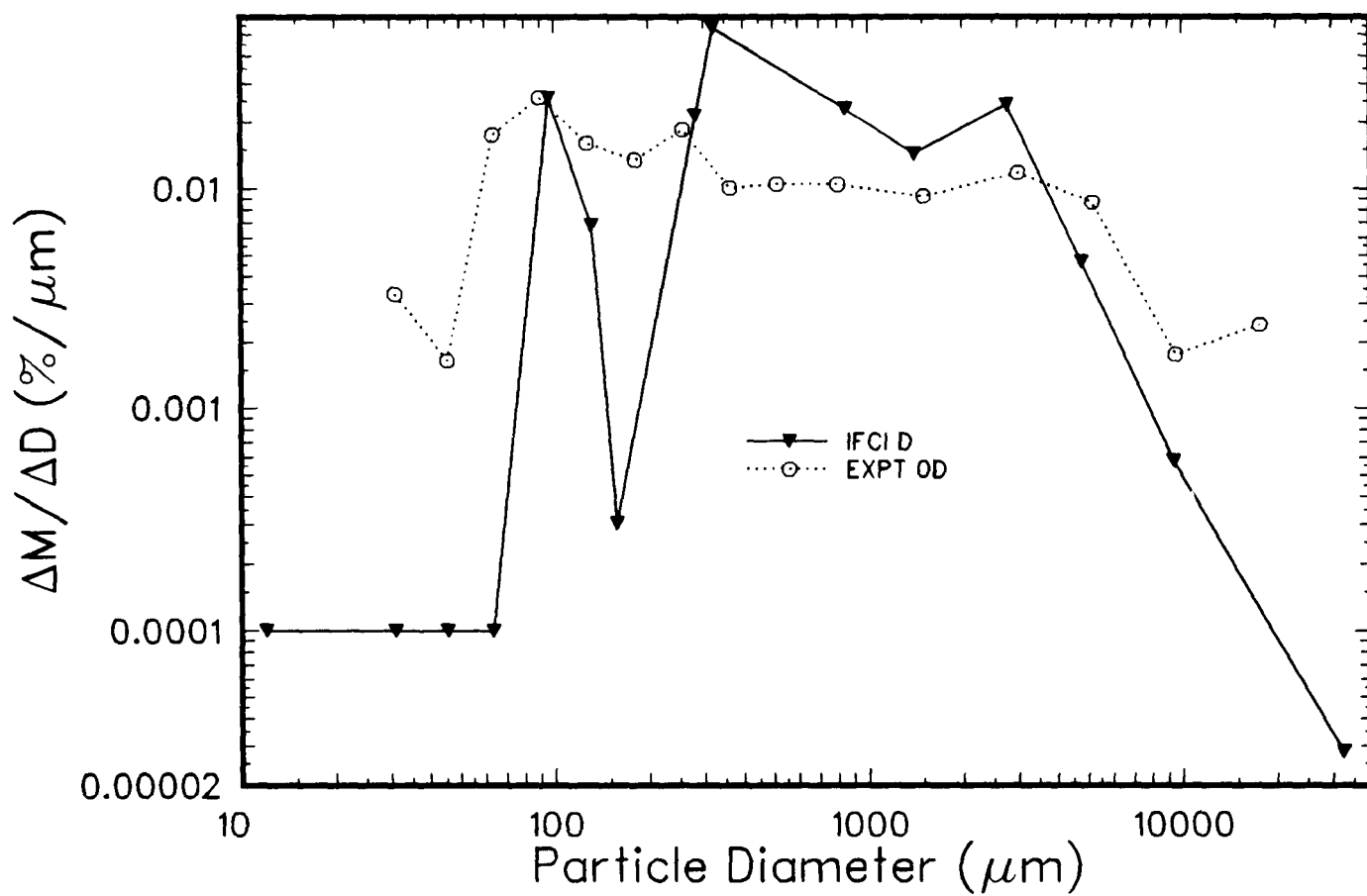


Fig. 8.9 Melt size distributions from IFCI problem and FITSOD posttest debris.

Results from the IFCI test are seen to be similar to those from the FITS nonexploding experiments which had an eruption, bearing out the observation that the IFCI result seemed to be a weak steam explosion. The FITS2D experiment, which was a nonexploder, has a Sauter mean twice that of the "erupters," plus (probably more significantly) a mass median four times that in the erupters. The single steam explosion experiment, FITS5D, has both a Sauter mean and mass median ten times smaller than the erupters.

8.8 DISCUSSION OF RESULTS

The results of the IFCI test calculation are very encouraging; the IFCI results are very similar to the experimental results from the FITS experiments. The shape of the melt as it falls through the water, the propagation speed of the pressure front through the melt, and the final melt size distribution all match those seen in the experiments. This would seem to indicate that IFCI is doing a good job of simulating the major phenomena occurring during the FCI process, particularly those operative during the coarse mixing phase. This gives confidence that IFCI can be used as a predictive tool for FCIs for which no experimental data is available, notably, those which may occur at reactor scale in a reactor geometry.

The present calculation represents the first simulation of the complete FCI process in more than one dimension and including dynamic fragmentation. Both two dimensions and dynamic fragmentation appear necessary for a code to correctly simulate the coarse mixing phase, and the presence of these two key features greatly aids the simulation of the propagation and expansion phases without having to make simplifying assumptions.

A steam chimney was observed to form during the coarse mixing phase, which lends credence to the hypothesis that the steam chimney is also present during actual experiments.

The ability of IFCI to provide detailed information about experimental features that are difficult or impossible to observe experimentally was demonstrated by, for instance, the calculation of the actual water/melt mass ratio from the IFCI output, versus that available from experimental observation [8.2].

8.9 FUTURE WORK

The IFCI test calculation revealed some modeling problems, which will be corrected in future versions of IFCI. One of the more noticeable ones is the bulk boiling model in MELPROG. This model introduces time delays between the introduction of melt into a cell and the commencement of boiling, which is somewhat noticeable during the coarse mixing phase in the lower cells of the problem, and is probably affecting the results significantly during the propagation phase. The magnitude of the propagated pressure front is also not as large as that observed during strong FCIs, which may be partially due to the use of the bulk boiling model. The weak propagation indicates the need for a fine fragmentation model that more closely represents the phenomena occurring during FCI propagation [8.2,8.16], notably entrapment and superheating of water.

The bulk boiling model will be replaced at the start of the next phase of IFCI development with a surface boiling model, probably patterned after that used by Bohl in his calculation of FCIs using SIMMER-II [8.17].

Future calculations will, first, repeat the present problem using a surface boiling model; the problem will then be repeated using a finer finite difference mesh. The next major step for IFCI will be a jet mixing calculation, probably using the EJET [8.18] tests for experimental comparison. The calculation of jet mixing is a crucial step for the eventual use of IFCI as a predictive tool in reactor safety, as the jet geometry is the more probable melt-water configuration in a reactor accident, rather than that of FITS, where a blob of melt is dropped into a tank of water.

8.10 REFERENCES

- 8.1 D.E. Mitchell, et al., Intermediate Scale Steam Explosion Phenomena: Experiments and Analysis, NUREG/CR-2145, SAND81-0124, Sandia National Laboratories, Albuquerque, NM (September 1981).
- 8.2 D. Jakeman and R. Potter, "Fuel-Coolant Interactions," Proceedings of the International Conference on Engineering of Fast Reactors for Safe and Reliable Operation, Karlsruhe, DDR (October 1972).

- 8.3 D.V. Swenson and M.L. Corradini, Monte Carlo Analysis of LWR Steam Explosions, NUREG/CR-2307, SAND81-1092, Sandia National Laboratories, Albuquerque, NM (October 1981).
- 8.4 M.G. Stevenson, Report of the Zion/Indian Point Study, NUREG/CR-1411, LA-8306-MS, Los Alamos National Laboratory, Los Alamos, NM (April 1980).
- 8.5 D.C. Williams, et al., "Impact of Chemical Phenomena in Direct Containment Heating," American Chemical Society Meeting, Anaheim, CA (September 1986).
- 8.6 A. Thyagaraja and D.F. Fletcher, Buoyancy-Driven, Transient, Two-Dimensional Thermo-Hydrodynamics of a Melt-Water-Steam Mixture, CLM-P790, UKAEA, Culham Laboratory, Abingdon, UK (1986).
- 8.7 R.E. Henry and H.K. Fauske, "Required Initial Conditions for Energetic Steam Explosions," Fuel-Coolant Interactions, HTD-V19, American Society of Mechanical Engineers (1981).
- 8.8 E.P. Hicks and D.C. Menzies, "Theoretical Studies on the Fast Reactor Maximum Accident," ANL 7120, pp.654-670 (1965).
- 8.9 P.J. Roache, Computational Fluid Dynamics, Hermosa Publishers, Albuquerque, NM (1972).
- 8.10 J.F. Dearing, "A Four-Field Model of PWR Degraded Cores," Proceedings of the Third International Topical Meeting on Reactor Thermal-Hydraulics, Newport, RI (October 1985).
- 8.11 M.F. Young, et al., "The FCI Potential of Oxide and Carbide Fuels: Results of the Prompt Burst Series at Sandia Laboratories," Fourth CSNI Specialist Meeting on Fuel-Coolant Interactions in Nuclear Reactor Safety, Bournemouth, UK (April 1979).
- 8.12 M.F. Young, "The TEXAS Code for Fuel-Coolant Interaction Analysis," Proceedings of the LMFBR Safety Topical Meeting, Lyon-Ecully, France (July 1982).
- 8.13 C.C. Chu and M.L. Corradini, "One-Dimensional Transient Model for Fuel-Coolant Fragmentation and Mixing Analysis," Proceedings of the International ANS/ENS Topical Meeting on Thermal Reactor Safety, San Diego, CA (February 1986).

- 8.14 M. Pilch, "Acceleration Induced Fragmentation of Liquid Drops," PhD Thesis, University of Virginia (1981).
- 8.15 M.L. Corradini and G.A. Moses, "A Dynamic Model for Fuel-Coolant Mixing," Proceedings of the International Meeting on LWR Severe Accident Evaluation, Cambridge, MA (1983).
- 8.16 L. Nelson, et al., "Photographic Evidence for the Mechanism of Fragmentation of a Single Drop of Melt in Triggered Steam Explosion Experiments," Journal of Nonequilibrium Thermodynamics (1986).
- 8.17 L.B. Luck, SIMMER-II: A Computer Program for LMFBR Disrupted Core Analysis, Rev. 1, NUREG/CR-0453, LA-7515-M, Los Alamos National Laboratory, Los Alamos, NM (July 1980).
- 8.18 B.W. Marshall, Jr. and M. Berman, "An Experimental Study of Isothermal and Boiling Liquid Jets," Proceedings of the 14th Water Reactor Safety Research Information Meeting (NUREG/CP-0082, vols. 1-6), Gaithersburg, MD (February 1987).

9.0 EJECT MODULE

The EJECT module is being developed to provide a link between the MELPROG code and the CONTAIN code [9.1] that performs a detailed analysis of the containment. This module will treat the release of debris from the vessel. Various vessel failure modes will be considered from gravity slump to high pressure melt ejection. For the latter, the EJECT module will calculate melt fragmentation by effervescence, atomization and hydrodynamic fragmentation. It then calculates the aerosol fraction and size distribution delivered to the containment atmosphere.

In addition to providing the flow regime data to the FLUIDS module, the EJECT module will provide information to the STRUCTURES module regarding the ablation of the hole in the vessel as the ejection process develops. The ablation rate of the hole is determined by balancing the convective energy transfer from the molten fuel to the energy required to heat and melt the steel. The model will be modified to account for the presence of a crust overlying the lower head.

Correlations that will be used by the EJECT module are discussed in the following sections. These models consider the flow of melt and gas through the hole. However, in an accident calculation it is possible that the molten fuel, solid debris and gas flow through the hole. Experiments are needed to determine if the presence of solid debris significantly alters these correlations.

9.1 Hole Ablation

The Zion Probabilistic Safety Study (ZPSS) [9.2] analysis concluded that rapid hole ablation begins shortly after the onset of melt discharge through a failed instrument-tube penetration in the reactor vessel. The ablation rate is determined by equating the convective energy transfer from the molten fuel to the energy required to heatup and melt the steel.

$$\frac{dD}{dt} = \frac{2h \Delta T}{\rho_w [C_{p,w}(T_{m,w} - T_w) + h_{f,w}]} \quad (9.1)$$

$$\Delta T = T_f - T_{m,f} \quad \text{ZPSS model} \quad (9.2)$$

$$\Delta T = T_f - T_{m,w} \quad \text{Current model} \quad (9.3)$$

Here D is the instantaneous diameter of the hole in the reactor vessel; T_f is the temperature of the molten fuel; $T_{m,f}$ is the melting temperature of the fuel; $T_{m,w}$ is the melting temperature of the wall (i.e., the wall of the pressure vessel which is steel); h is the heat transfer coefficient from the molten fuel; ρ_w is the density of the pressure vessel; $C_{p,w}$ is the specific heat of the wall; and $h_{f,w}$ is the heat of fusion of the wall.

The ZPSS analysis further concluded that a thin crust (86 μm in the supporting calculation) insulated the ablating steel from the molten fuel; consequently, the temperature difference between molten and solidified fuel (Eq.(9.2)) governs ablation. According to the ZPSS model, hole ablation will not occur unless the melt is superheated. We feel that such a thin crust is not stable and cannot prevent molten fuel contact with molten steel; therefore, the temperature difference between fuel and molten steel (Eq.(9.3)) governs the hole ablation [9.3].

The small length-to-diameter ratio (L/D) of the ablating hole coupled with rapid ablation dictates careful treatment of the heat transfer coefficient in Eq.(9.1). Development of the thermal and hydrodynamic boundary layers is incomplete (boundary layer thickness less than hole radius) for the small L/D ratio associated with an ablating hole in a reactor vessel. Consequently, the heat transfer coefficient is larger than would be expected for fully developed flow in a long tube. The ZPSS analysis does not consider this entrance region effect. We recommend choosing the larger value of h as determined from a flat plate correlation and from a tube correlation for fully-developed turbulent flow. Our experience, however, is that the flat rate correlation is almost always the appropriate choice for both experiments and reactor analysis.

Flat plate correlation [9.4]:

$$h = 0.0292 \frac{k_L}{L} \text{Re}_L^{0.8} \text{Pr}^{0.33} \quad (9.4)$$

Tube correlation [9.4]:

$$h = 0.023 \frac{k_L}{D} \text{Re}_d^{0.8} \text{Pr}^{0.33} \quad (9.5)$$

Here h is the local heat transfer coefficient at the breach exit; k_L is the thermal conductivity of the molten fuel; Re_L is the Reynolds number based on the length (L) of the hole; Re_d is the

Reynolds number based on the instantaneous diameter (D) of the hole; and Pr is the Prandtl number of the molten fuel.

Actually, the heat transfer coefficients given by Eqs.(9.4) and (9.5) are appropriate only for nonablating surfaces. If the ablation rate (D) is large, then the heat transfer coefficient is reduced because of 'transpiration cooling' of the boundary layer [9.5]. The modified coefficient is given by

$$h = \frac{\beta}{e^{\beta}-1} \quad (9.6)$$

where

$$\beta = \frac{\rho_{mw} C_{p,mw} \dot{D}}{2h} \quad (9.7)$$

Here ρ_{mw} and $C_{p,mw}$ are the density and specific heat of the molten wall material (steel); and D is the ablation rate (i.e., the rate change of hole diameter).

The effective heat transfer coefficient is significantly reduced when gas blowthrough occurs. Only the molten fuel component of the two phase mixture passing through the hole is capable of ablating the steel; the blowdown gas (steam and hydrogen) is not hot enough (.600 K) to melt the pressure vessel wall. Following gas blowthrough, we reduce the heat transfer coefficient by the relative amount of molten fuel that is in contact with the steel.

$$h = h \frac{A_L}{A_h} \quad (9.8)$$

Here A_h is the area of the hole in the pressure vessel.

Hole ablation has been observed in experiments at SNL where a molten mixture of Fe and Al_2O_3 ($T_f=2800$ K) was ejected under pressure from a steel pressure vessel ($T_w=300$ K). Pilch and Tarbell [9.3] report excellent agreement between the data and the above model.

9.2 Gas Blowthrough

During the process of withdrawing fluids from tanks, surface distortion and subsequent gas ingest has been observed in situations where no liquid rotation have been present [9.6]. The

distortion takes the form of a funnel-like depression of the liquid surface over the outlet hole, which leads to blowthrough of gas in the outlet before all of the liquid has drained from the tank. This phenomena becomes more important as the flow rate out the hole increases.

Consider a very deep pool of liquid draining through a hole. Initially, the free surface of the liquid pool remains level as the pool drains. However, surface distortion and gas blowthrough develop very rapidly when the pool has drained down to some critical depth (H'). Based on experiment measurements of H' in cylindrical and hemispherical tanks, Gluck [9.6] proposed the following empirical correlation, which we use in our work.

$$H' = 0.43 \left(\frac{4}{\pi A_s} \right)^{0.5} \tanh \left[\left(\frac{A_L}{A_s} \right)^{0.5} Fr^{0.25} \right] \quad 10^2 < Fr < 10^6 \quad (9.9)$$

$$Fr = \frac{V_L^2}{g \left(\frac{4}{\pi A_L} \right)^{1/2}} \quad \text{Froude number} \quad (9.10)$$

where A_L is the area occupied by the liquid flowing through the hole, A_s is the area of the free surface of the pool, V_L is the liquid discharge velocity and g is the gravitational acceleration. Careful examination of Eq.(9.9) shows that H' is independent of A_s (for a fixed flow rate) when $A_s/A_L > 400$. The critical depth (H') decreases for smaller values of A_s/A_L ; thus, gas blowthrough is delayed in smaller tanks that otherwise have the same hole size and discharge rate.

The expected range of Froude numbers for the SNL experiments and the reactor accidents are 10^3 to 10^4 , within the applicable range of Gluck's correlation. Keep in mind that prior to gas blowthrough $A_L = A_h$.

Equation (9.10) indicates that the critical depth for gas blowthrough (H') is calculated given the liquid flow area ($A_L = A_h$) inside the hole. We reverse the logic following gas blowthrough. Given the instantaneous pool depth (H), we calculate the area of liquid flow (A_L) inside the hole as shown below in Eq.(9.11). This ensures that flow rate of liquid out the hole is sufficient to satisfy the condition for gas blowthrough.

$$A_L = A_h \left[\frac{\operatorname{arctanh} \left(\frac{H}{0.43 \left(\frac{4}{\pi} A_s \right)^{0.5}} \right)}{\frac{A_h}{A_s}^{0.5} Fr^{0.25}} \right]^{8/3} \quad (9.11)$$

$$Fr = \frac{V_L^2}{g \left(\frac{4}{\pi} A_h \right)^{0.5}} \quad (9.12)$$

9.3 Jet Disruption

Jet disruption during melt discharge occurs by two main processes: effervescing gas and pneumatic atomization. Jet disruption by effervescing gas occurs when hydrogen rapidly comes out of solution with the steel component of molten core debris. Pneumatic atomization occurs following gas blowthrough. Here the relative velocity between gas and liquid in the vessel breach is responsible for liquid atomization. These atomization processes can be sufficiently violent so as to be a potential source of aerosols. In addition, some fraction (i.e., the small particle portion of the particle size distribution) of the atomized debris may be carried directly out of the cavity without ever impacting surfaces of the cavity.

The violent disruption of jets by effervescing gas is a process that has been exploited commercially by the steel industry [9.7-9.9]. Vacuum degassing of steel melts is a process that removes unwanted hydrogen from molten steel by teaming the melt into a vacuum chamber. The effervescing hydrogen violently fragments the jet into particles of a few hundred microns in size, and the large surface area of the particles allows for the rapid diffusion of the remaining hydrogen from the melt.

This process is quite similar to what can happen in a reactor accident. Powers has shown that hydrogen rapidly goes into solution with the steel component of molten core debris and that the solubility of hydrogen in steel is quite high at the elevated temperatures and pressures that could exist in the primary system prior to vessel failure.* The effervescence of hydrogen from the

* D.A. Powers, "Solubility of Gases in Molten Materials Produced During Reactor Core Meltdown," SAND83-1796, Sandia National Laboratories, Albuquerque, NM (to be published).

molten core debris will violently fragment the jet when melt is ejected into the low pressure environment in the reactor cavity. Disruption of thermite jets by the effervescence of nitrogen gas has been observed in experiments at SNL [9.10-9.12]. The observed jet disruption and aerosol production are indicative of what can happen in the reactor accident because the solubility of N_2 in the iron phase of thermite; is similar to the solubility of H_2 in the steel phase of molten core debris.

A mechanistic model does not exist currently that can predict the observed particle sizes produced by effervescing gas. We note, however, that the solubility of H_2 in steel is about a factor of 5 larger in the reactor accident when compared to typical vacuum degassing processes. This suggests that jet disruption might be more energetic in the reactor accident, particularly when molten steel is stratified from the more dense constituents of core debris. On the other hand, molten core debris is less than 20% steel, thus compensating for the higher solubility if the molten steel is well mixed with the remainder of the melt. Extrapolation of observations from vacuum degassing to the reactor accident becomes even more difficult because the jet diameter in the reactor accident could be 10-50 times larger than those commonly used in vacuum degassing. Quantitative predictions or extrapolations cannot be made until a mechanistic model, which is being developed at SNL, becomes available.

Pneumatic atomization occurs when liquid and gas are simultaneously accelerated through an orifice. The liquid is atomized by the gas because of the difference in relative velocities as each phase passes through the orifice. Although pneumatic atomizers have been produced commercially and studied extensively since the early 1900's, there is no universally accepted correlation (or theory) for the observed particle sizes. However, the literature does tell us that the distribution of particles sizes is log-normal with a mass median size (d_m) and a geometric standard deviation (σ_g). We select three correlations for comparison from the dozens that appear in the literature. We chose these correlations because they are dimensionless and because each contains the important ratio of gas-to-liquid flow rate.

Deyson correlation [9.13]:

$$\frac{d_m}{d_L} = \frac{8 \left[1 + 2.75 \left(\frac{\dot{m}_L}{\dot{m}_g} \right)^{0.66} \right]}{\left[\frac{\rho_L}{\rho_g} \frac{d_L}{d_g} We \right]^{1/2}} \quad (9.13)$$

Lubanska correlation [9.14]:

$$\frac{d_m}{d_L} = 50 \left[\frac{1}{We} \left(\frac{\rho_g}{\rho_L} \right)^2 \frac{\mu_L}{\mu_g} \left(1 + \frac{\dot{m}_L}{\dot{m}_g} \right) \right]^{1/2} \quad (9.14)$$

Wigg correlation [9.15]:

$$\frac{d_m}{R_g} = 200 \frac{\left(\frac{\rho_g}{\rho_L} \right)^{0.2} On^{0.6} Re^{0.1}}{We^{0.5}} \left[1 + \frac{\dot{m}_L}{\dot{m}_g} \right]^{1/2} \quad (9.15)$$

The following definitions apply to the above equations.

$$d_L = \left[\frac{4}{\pi} A_L \right]^{1/2} \quad (9.16)$$

$$d_g = \left[\frac{4}{\pi} (A_h - A_L) \right]^{1/2} = 2R_g \quad (9.17)$$

$$We = \frac{\rho_g (V_g - V_L)^2 R_g}{\sigma} \quad \text{Weber number} \quad (9.18)$$

$$We' = \frac{\rho_g (V_g - V_L)^2 R_g}{\sigma} \quad (9.19)$$

$$On = \frac{\mu_L}{[\rho_L \sigma R_g]^{0.5}} \quad \text{Ohnesorge number} \quad (9.20)$$

$$Re = \frac{\dot{m}_L}{R_g \mu_L} \quad \text{Reynolds number} \quad (9.21)$$

Here d_L and d_g are the equivalent diameter of the liquid and gas streams in the orifice; R_g is the gas core radius or the gas annulus thickness; μ_L and μ_g are the viscosities of the liquid and the gas; and σ is the liquid surface tension.

Reported values [9.14,9.16,9.17] of the geometric standard deviation (σ_g) for pneumatic atomization range from 1.4 (i.e., a very narrow distribution) to 10 (i.e., a very broad distribution). Lubanska [9.14], Hugo [9.16], and Gretzinger [9.17] have correlated σ_g with the mass median particle size (d_m expressed in microns). These correlations are not dimensionless. Furthermore, Lubanska and Hugo report conflicting results; Lubanska found that σ_g increases with increasing d_m while Hugo found that σ_g decreased with increasing d_m . We conclude that reasonable estimates of σ_g cannot be made at this time.

Each of the above equations shows a different dependence on length scale and properties of the liquid and gas; consequently, we recommend caution when scaling up from the comparatively small devices on which the correlations are based to the full reactor scale.

9.4 References

- 9.1 K.D. Bergeron, et al., User's Manual for CONTAIN 1.0, A Computer Code for Severe Nuclear Reactor Accident Containment Analysis, NUREG/CR-4085, SAND84-1204, Sandia National Laboratories, Albuquerque (May 1985).
- 9.2 "Zion Probabilistic Safety Study," Commonwealth Edison Co., Chicago, IL (Sept. 1981).
- 9.3 M. Pilch and W.W. Tarbell, High Pressure Ejection of Melt From a Reactor Pressure Vessel: The Discharge Phase, NUREG/CR-4383, SAND85-0012, Sandia National Laboratories, Albuquerque, NM (September 1985).
- 9.4 A.J. Chapman, Heat Transfer, 3rd ed., Macmillan Publishing Co., Inc. (1974).
- 9.5 A.J. Suo-Anttila, "Simplified Multicomponent Phase Transition Model," LA-7557-MS, Los Alamos National Laboratory, Los Alamos, NM (Jan. 1979).
- 9.6 D.F. Gluck, J.P. Gille, E.E. Zukoski, and D.J. Simkin, "Distortion of a Free Surface During Tank Discharge," J. Spacecraft and Rockets, 3(1), pp. 1691-1692 (Nov. 1966).

- 9.7 A. Ogunleye, "Removal of Dissolved Hydrogen in Liquid Iron," Second International Congress on Hydrogen in Metals, Paris, France (June 6-11, 1977).
- 9.8 J.A. Belk, Vacuum Techniques in Metallurgy, Pergamon Press, 63 (1963).
- 9.9 N.A. Warner, "Stream Breakup in Vacuum Degassing," J. Iron and Steel, pp. 44-50 (Jan. 1969).
- 9.10 W.W. Tarbell, J.E. Brockmann, and M. Pilch, High-Pressure Melt Streaming (HIPS) Program Plan, NUREG/CR-3025, SAND82-2477, Sandia National Laboratories, Albuquerque, NM (August 1984).
- 9.11 W.W. Tarbell, M. Pilch, and J.E. Brockmann, "Behavior of Core Debris Ejected from a Pressure Vessel Into Scaled Reactor Cavities," Proceedings of the 12th Water Reactor Safety Information Exchange Meeting (NUREG/CP-0058, vols. 1-6), Gaithersburg, MD (January 1985).
- 9.12 J.E. Brockmann, W.W. Tarbell, "Aerosol Source Term in High Pressure Ejection," Nuc. Sci. and Eng., 88, pp. 342-356 (1984).
- 9.13 J.Y. Deysson and J. Karian, "Approximate Sizing of Single Fluid and Pneumatic Atomizers," Proceedings of the 1st International Conference on Liquid Atomization and Spray Systems, Tokyo, Japan, p. 243 (Aug. 27-31, 1978).
- 9.14 H. Lubanska, "Correlation of Spray Ring Data for Gas Atomization of Liquid Metals," J. of Metals, pp. 45-48 (Feb. 1980).
- 9.15 L.D. Wigg, "Drop-Size Prediction for Twin-Fluid Atomizers," J. Inst. Fuel, pp. 500-505 (Nov. 1964).
- 9.16 P.E. Hugo and R.M. German, "Characteristics of Tin Powder Produced by an Annular Gas Atomization Nozzle," Int. J. Powder Metallurgy and Powder Technology, 18(4), pp. 301-311 (1982).
- 9.17 J. Gretzinger and W.R. Marshall, Jr., "Characteristics of Pneumatic Atomization," AIChE J., 7(2), pp. 312-318 (June 1961).

10.0 MATERIAL PROPERTIES

The thermophysical properties of all the materials that are modeled in MELPROG except hydrogen and water are determined in the MELPRO subroutine. In addition to properties for pure materials, MELPRO determines properties for mixtures of certain materials that may form during core melt. In general, all properties are functions of temperature; additionally, mixture properties are functions of composition. The temperatures of all components in a mixture are assumed to be equal, and all properties are assumed to be isotropic. Presently, the properties included are thermal conductivity, specific heat, specific enthalpy, density and viscosity. Additionally, given specific enthalpy, MELPRO returns temperature. Materials included in MELPRO are uranium dioxide, zircaloy, zirconium dioxide, 304 and 316 stainless steel, inconel 600 and 718, medium carbon steel A508, silver-indium-cadmium control alloy and boron carbide.

10.1 Detailed Description

Properties for UO_2 , zircaloy, ZrO_2 , 304 ss, Ag-In-Cd alloy and B_4C were extracted from Refs. [10.1] and [10.2]. Properties for Inconel 600, Inconel 718 and medium carbon steel A508 came from Ref. [10.3], and properties for 316 ss came from Ref. [10.4].

The MELPRO parameter list (IMAT, IPROP, TX, LT, CX, LC, X) usually contains all of the input required by the subroutine. The only exception occurs when material porosity must be calculated, in which case, the common block CRQUIV is used. This special case will be discussed later. IMAT is the material number identifier, and Table 10.1 lists the material associated with each value of IMAT. IPROP is the property indicator number, and Table 10.2 lists the meaning of the parameters in the parameter list for each acceptable value of IPROP.

Figure 10.1 is a flow chart of the MELPRO subroutine. Pure materials and mixtures are treated separately. The composition of mixtures is identified by the IEUT(k) and FCOMP(k) arrays; IEUT(k) is the material number identifier and FCOMP(k) is the mass fraction of the k^{th} component in the mixture. For pure materials IEUT(1)=IMAT and FCOMP(1)=1. LC<0 in the parameter list indicates that the CX array contains the relative mass fractions of the |LC| components defined for material IMAT, so it is only necessary to normalize the CX array to unity to obtain

Table 10.1

Materials Treated by MELPRO

<u>IMAT</u>	<u>Material</u>
1	UO ₂
2	- ² (currently undefined)
3	Zircaloy
4	ZrO ₂
5	304 ² stainless steel
6	-
7	316 stainless steel
8	-
9	-
10	-
11	medium carbon steel A508
12	-
13	Inconel 718
14	-
15	Inconel 600
16	-
17	Ag-In-Cd alloy
18	-
19	B ₄ C
20-50	-
51	U-Zr-O-304ss "eutectic" (1)
52	Ag-In-Cd-Zr-O-304ss "eutectic" (2)
53	U-Zr-O-304ss-Ag-In-Cd "eutectic" (3)

Notes

1. currently modeled as a mixture of UO₂, Zr, ZrO₂ and 304ss components
2. currently modeled as a mixture of Ag-In-Cd alloy, Zr, ZrO₂ and 304ss components
3. currently modeled as a mixture of UO₂, Zr, ZrO₂, 304ss and Ag-In-Cd alloy components

Table 10.2 Properties Calculated by MELPRO

IPROP	TX,LT(1)	CX,LC (2)	X
1	temperature (K)	mass or mass fraction	thermal conductivity (W/m-K)
2	temperature (K)	mass or mass fraction	specific heat capacity (J/kg-K)
3	temperature (K)	mass or mass fraction	specific enthalpy (J/kg)
4	temperature (K)	mass or mass fraction	density (kg/m ³)
5	temperature (K)	mass or mass fraction	viscosity (kg/m-s)
6		(currently not used)	
7		(currently not used)	
8	specific enthalpy (J/kg)	mass or mass fraction	temperature (K)
9-17		(currently not used)	
18	dummy	mass	mixture mass fraction
19	temperatures (K) (3)	mass or mass fraction	phase transition parameters (4)
20	dummy	mass or mass fraction	solidus-liquidus temperatures (K)
21	temperature (K)	mass or mass fraction	thermal conductivity
			specific heat capacity
			phase transition parameters
25	dummy		solidus-liquidus temperatures
			solidus-liquidus enthalpies
29	temperature (K)	mass or mass fraction	specific heat capacity
			thermal conductivity
			density
			specific enthalpy
			viscosity
35			mixture mass fraction
			specific enthalpy
38			thermal conductivity
			specific heat capacity

NOTES

- 1) It is the length (# of consecutive addresses) of the TX array; LT<0 indicates that the material porosity should be considered.
 - 2) LC is the length of the CX array; LC<0 indicates that the CX values are the relative mass fractions of the various components that define mixture IMAT, LC>0 indicates that the CX values are relative masses of the various elements in the mixture defined by IMAT.
 - 3) for IPROP=19, MELPRO will search for phase transition points between temperatures TX(1) and TX(2) for material IMAT
 - 4) X(1) = lower transition limit (K)
X(3) = phase transition enthalpy/(X(2)-X(1))
- X(2) = upper transition limit (K)
X(4) = solidus temperature (K)

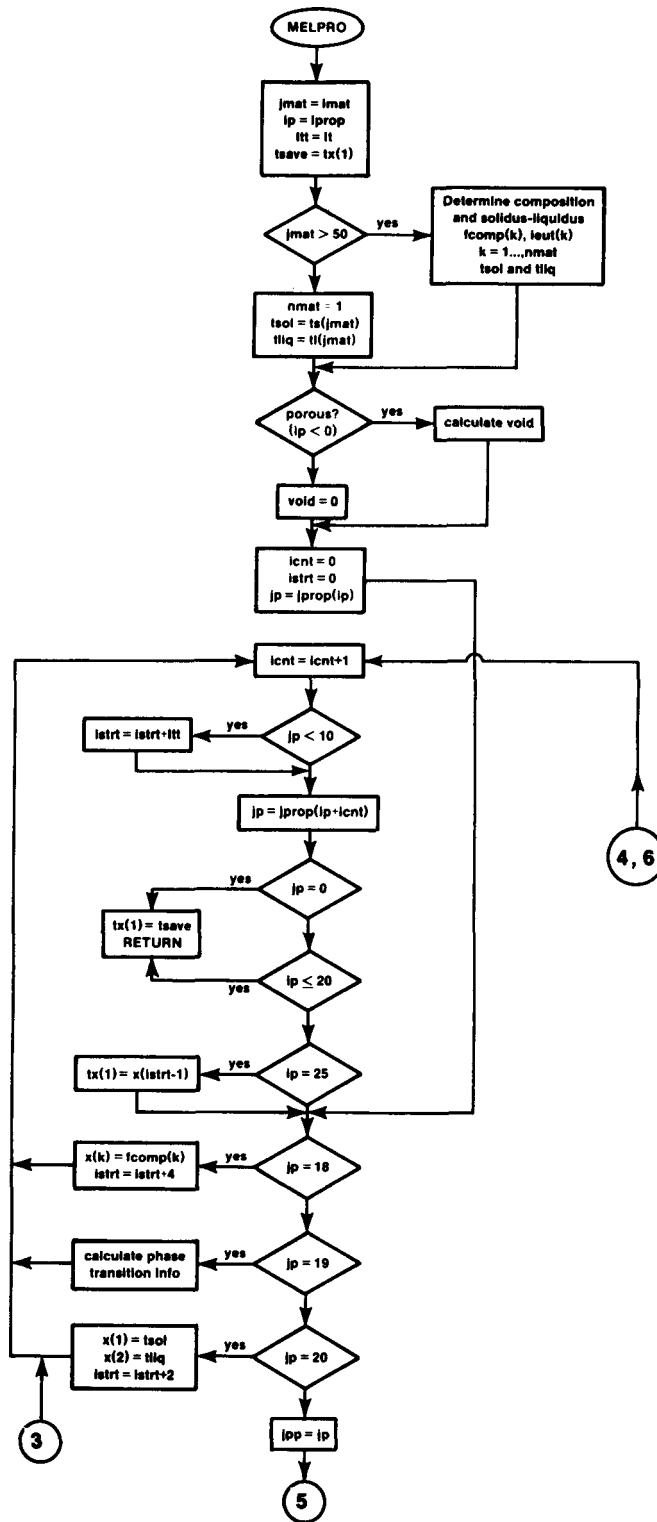


Fig. 10.1a Flow chart showing structure of the MELPRO subroutine.

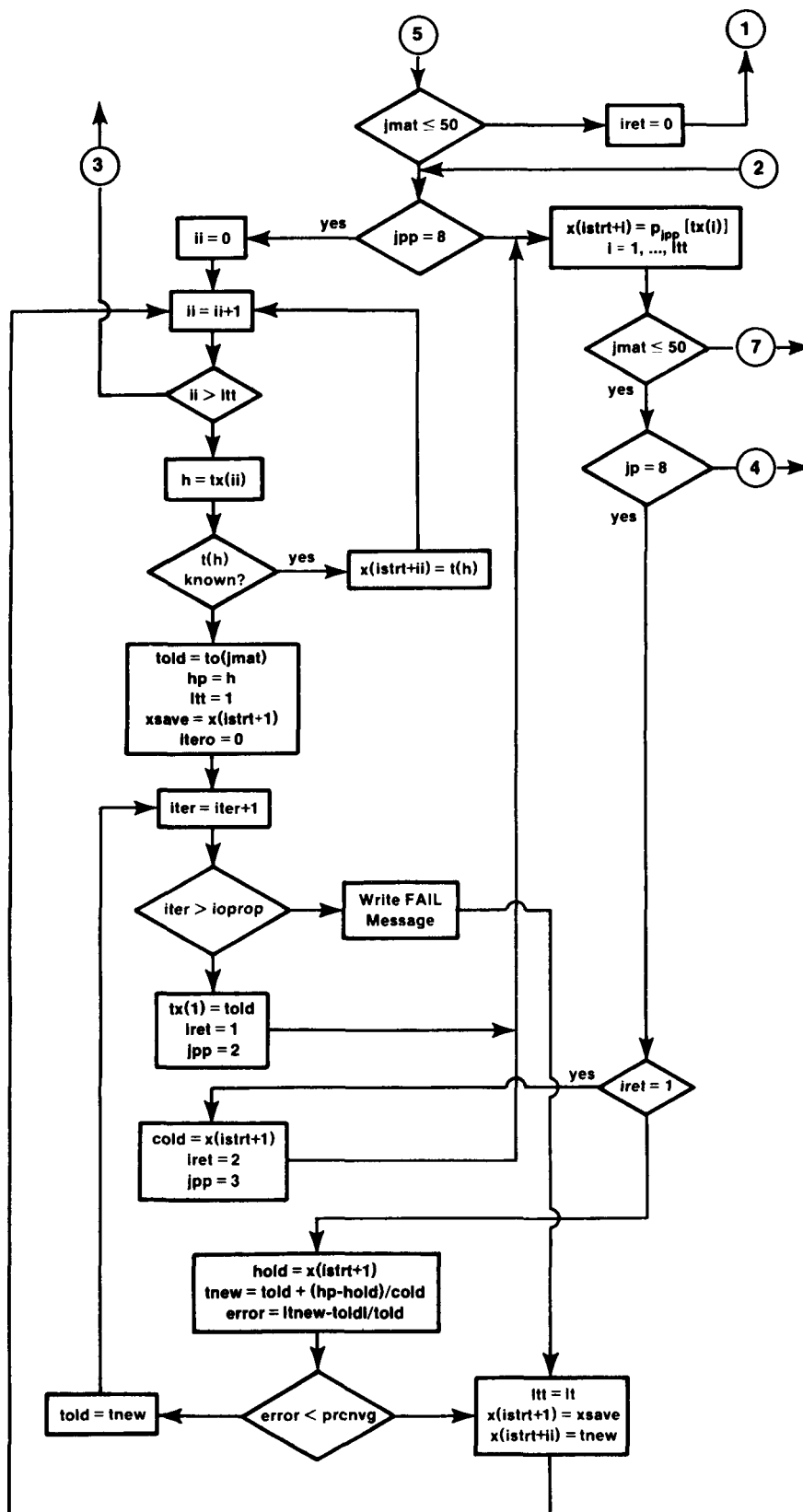


Fig. 10.1b Flow chart showing structure of the MELPRO subroutine. (continued)

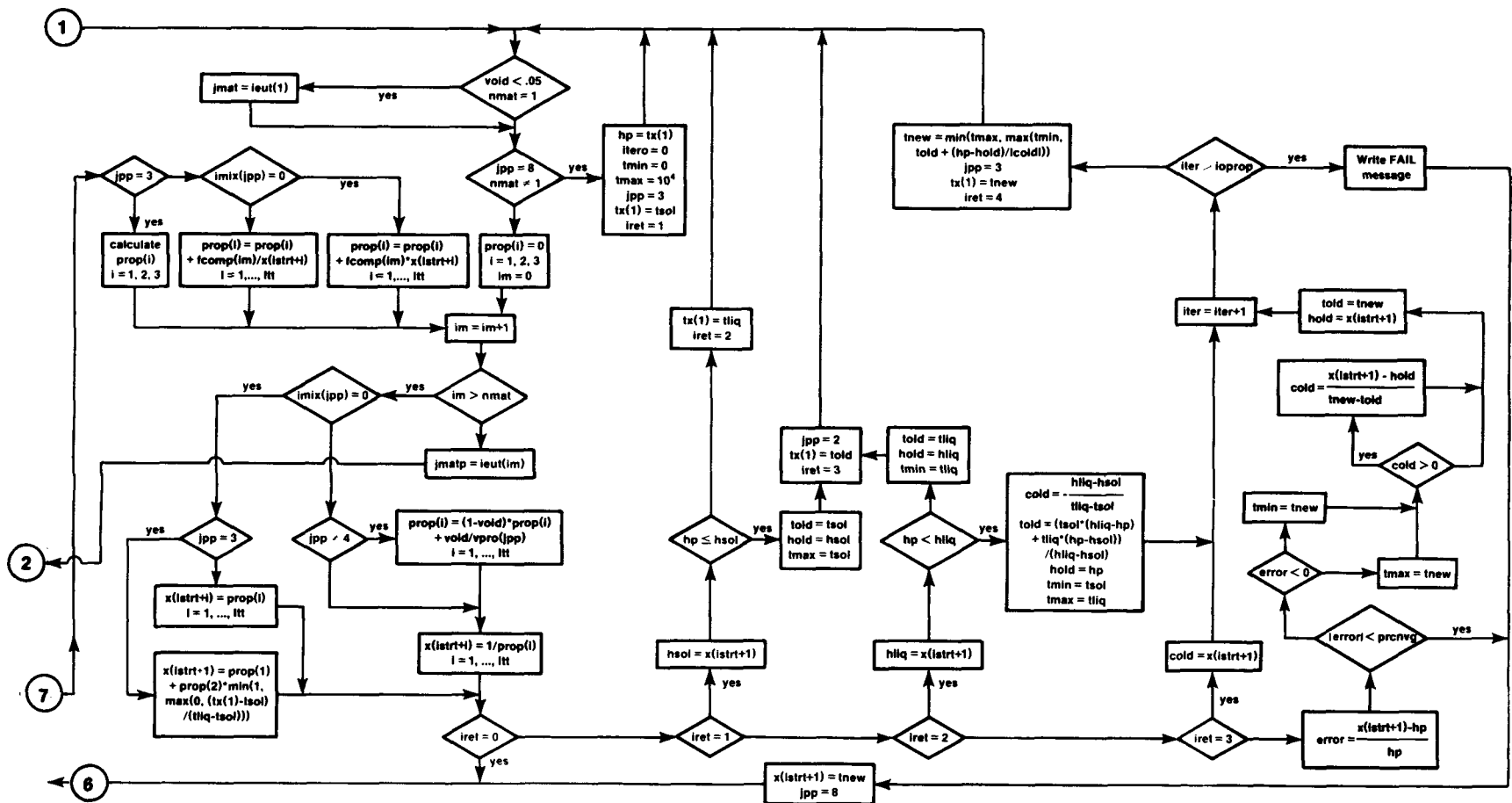


Fig. 10.1c Flow chart showing structure of the MELPRO subroutine. (continued)

FCOMP(k), $k=1, \dots, |LC|$. $LC > 0$ indicates that the CX array contains the masses of individual "elements" that material IMAT contains (e.g., U, Zr, O and 304ss for IMAT=51), in which case it is necessary to calculate the mass fractions of the various components according to certain prescribed rules listed in Table 10.3 (e.g., the mass fractions of UO_2 , Zr, ZrO_2 and 304ss for IMAT=51).

ICNT is the property counter that is used to increment the property indicator array pointer, IPROP, when multiple properties are returned. MELPRO returns to the calling routine when the pointer reaches a value of JP=0 in the property indicator array or after it has calculated a single property when $IPROP \leq 20$. ISTRT is the property array pointer that indicates where to write the next property in the output array X; ISTRT is usually incremented by LT after each property has been determined for LT values of temperature.

Properties identified with $10 < JP \leq 20$ are determined in the flow chart using generalized algorithms with material specific array data. When JP=19, MELPRO searches for phase transition points for material IMAT that lie between temperatures TX(1) and TX(2) specified in the parameter list. The algorithm is complicated by the MELPRO convention for treating phase transition points in a mixture. The convention will be discussed in more detail when specific enthalpy of mixtures is addressed below. Briefly, however, the heat of fusion for a mixture occurs between the solidus and liquidus temperatures defined for that particular composition irrespective of the locations of the individual solidus-liquidus points of the components. Additionally, any other component phase transitions or portions thereof that occur above the mixture solidus temperature are lumped into the mixture heat of fusion. The algorithm determines the heat of fusion and other phase transition points (range of temperature) and compares the range to the given TX(1) and TX(2). If there is any overlap, the point is included. MELPRO returns four values. The first value is an effective specific heat capacity, which is equal to the sum of all the included phase transition enthalpies divided by the temperature difference between the maximum temperature in the included ranges and the minimum. The second value is the minimum temperature, the third value is the maximum temperature and the fourth value is the solidus temperature.

For $0 < JP \leq 10$ the routine branches to either the appropriate

Table 10.3

Rules for Determining Mixture Compositions

IMAT=51:

$$CX, LC = m_U, m_{Zr}, m_O, m_{304ss}$$

$$LC=4$$

$$CX, LC = f_{UO_2}, f_{Zr}, f_{ZrO_2}, f_{304ss}$$

restrictions

all U exists as UO_2 all O exists as UO_2 or ZrO_2

no 304ss oxide currently

IMAT=52:

$$CX, LC = m_{Ag-In-Cd}, m_{Zr}, m_O, m_{304ss}$$

$$LC = 4$$

$$CX, LC = f_{Ag-In-Cd}, f_{Zr}, f_{ZrO_2}, f_{304ss}$$

$$LC = -4$$

restrictions

all Ag-In-Cd exists as a single species

all O exists as ZrO_2

no 304ss oxide currently

IMAT=53:

$$CX, LC = m_{UO_2}, m_{Zr}, m_{ZrO_2}, m_{304ss}, m_{Ag-In-Cd}, m_U, m_{Zr}, m_O$$

$$LC=8$$

$$CX, LC = f_{UO_2}, f_{Zr}, f_{ZrO_2}, f_{304ss}, f_{Ag-In-Cd}$$

$$LC=5$$

restrictions

all U exists as UO_2 all O exists as UO_2 or ZrO_2

all Ag-In-Cd exists as single species

no 304ss oxide currently

pure material section (IMAT≤50) or to a generalized mixture section (IMAT>50). The calculation of pure material properties other than temperature from enthalpy is straight-forward, and following this MELPRO increments ICNT and ISTRT and moves to the next property specified in the property indicator array. For temperature from enthalpy (JP=8) for a pure material, the routine will branch to a generalized specific enthalpy function inversion algorithm if the form of the enthalpy function is too complicated (transcendental or polynomial of third order or higher); otherwise, it will calculate the temperature directly from the inverted enthalpy function. The generalized inversion algorithm is iterative and involves using the formulas for the material's specific heat capacity and specific enthalpy to solve

$$TNEW = TOLD + (HP-HOLD)/COLD$$

where

HP = the given specific enthalpy,
TNEW = new iterate value of temperature,
TOLD = old iterate value of temperature,
COLD = specific heat capacity at TOLD, and
HOLD = specific enthalpy at TOLD

until

$$ERROR = \frac{|TNEW - TOLD|}{TOLD} < PRCNVG$$

where PRCNVG is the convergence criteria. The iterative algorithm is depicted in Figure 10.1.

The calculation of mixture properties other than temperature from enthalpy involves calculating and summing mass or volume weighted properties for the individual components. Properties such as specific heat capacity and specific enthalpy that are expressed on a per unit mass basis are mass weighted as follows

$$PROP_{mix} = \sum_k FCOMP(k) * PROP(IEUT(k))$$

while properties such as density that are expressed on a volumetric basis are weighted as follows

$$PROP_{mix}^{-1} = \sum_k FCOMP(k) / PROP(IEUT(k))$$

The calculation of specific enthalpy is actually more complicated than suggested above. If the formula above was used without modification, the definition of the solidus-liquidus point would become unclear because that formula indicates that the mixture undergoes k different solidus-liquidus transitions between the solidus and liquidus temperature of each component. Actually, a mixture usually behaves more like a single material with a composition dependent solidus and liquidus temperature. MELPRO determines a composition dependent solidus and liquidus temperature for each mixture immediately after ascertaining the composition. For IMAT=51 (U-Zr-O-304ss) the temperatures are calculated from pseudo-binary phase diagrams for $\text{UO}_2\text{-Zr}$, Zr-ZrO_2 and $\text{UO}_2\text{-ZrO}_2$ and the solidus and liquidus temperatures for 304ss. For IMAT=52 and IMAT=53 the temperatures are presently just read from data statements. The mixture heat of fusion is equal to the sum of the mass weighted component heats of fusion plus the portion of any other phase transitions that occur above the mixture solidus temperature. Therefore, only that fraction of the calculated mixture heat of fusion given by

$$\text{HFUS}_{\text{mix}} * \max\left[0, \min\left(1, \frac{T - \text{TSOL}}{\text{TLIQ} - \text{TSOL}}\right)\right]$$

is included in the calculated specific enthalpy at temperature T. This implies that the sum of the mass weighted specific enthalpies at a given temperature is not necessarily equal to the specific enthalpy of the mixture calculated at the same temperature. The calculation of temperature from enthalpy for a mixture is similar to the iterative calculation for pure materials discussed above. The primary difference between the two algorithms is that the mixture algorithm has a few preliminary steps to determine where the given specific enthalpy, HP, lies with respect to the solidus and liquidus enthalpies of the given composition, HSOL and HLIQ, respectively. If HP lies between HSOL and HLIQ, then the effective specific heat capacity is given by

$$\text{COLD} = \frac{\text{HLIQ} - \text{HSOL}}{\text{TLIQ} - \text{TSOL}}$$

independent of the iterate TOLD. Otherwise, at each iteration COLD is given by

$$\text{COLD} = \frac{H(\text{TNEW}) - H(\text{TOLD})}{\text{TNEW} - \text{TOLD}} .$$

10.2 References

- 10.1 D.L. Hagrman, Materials Properties for Severe Core Damage Analysis, EGG-CDD-5801, EG&G, Idaho Falls, ID (May 1982).
- 10.2 D.L. Hagrman, G.A. Reymann and R.E. Mason, MATPRO-Version 11 (Revision 2) A Handbook of Materials Properties for Use in the Analysis of Light Water Reactor Fuel Rod Behavior, NUREG/CR-0497 and TREE-1280, Revision 2 (August 1981).
- 10.3 Nuclear Systems Material Handbook, Vols. 1-4, TID-26666.
- 10.4 Properties for LMFBR Safety Analysis, Argonne National Laboratory, Argonne, IL (1976).

A.0 CODE STRUCTURE

This appendix describes the functional layout and operation of the following MELPROG modules: FLUIDS, CORE, STRUCTURES, DEBRIS and RADIATION. As discussed earlier, the VICTORIA, IFCI and EJECT modules are in the developmental stage: the implementation of these modules will be described in later reports. For each module, the purpose and operation of the main subroutines are described. In addition to describing the modules, the TRAC-MELPROG link, the initialization section and the data storage system are also discussed. Figure A.1 illustrates the overall structure of MELPROG.

The main MELPROG driver calls each of the module controlling subroutines. At the beginning of a problem, the initialization routine MPRGIN is called. Then, on each time step the other modules are called in the following order: FLUIDS (FLDYN), CORE (CORE), STRUCTURES (STRUCT), DEBRIS (DEBRIS), and RADIATION (RADMIM). The driver also checks for end of problem time, performs graphical output on CMOUT by calling GRF every GFINT time steps, and writes restart dumps at DMPINT intervals. The input for a problem should be stored on a file named MELIN. The output from a problem will appear on a file MELOUT. If a restart is performed, the dump file MELDMP should be renamed MELRST.

A.1 Fluids Module

This section describes the functional layout, purpose and operation of the FLUIDS module. The functional arrangement of the subroutines in the FLUIDS module is illustrated in Fig. A.2. The FLUIDS module is entered through the FLDYN subroutine. This subroutine assembles volumetric mass and energy sources from the other modules. For example, hydrogen mass sources and energy sources are provided by the CORE, DEBRIS and STRUCTURES modules to account for the oxidation of zirconium and stainless steel.

The FLDYN subroutine calls the FLDYNA subroutine. FLDYNA, in turn, calls two subroutines: FLUIDS, which is the run-time driver for the fluid dynamics calculation, and FLPRNT, which prints the results of the calculation at specified times. The FLUIDS subroutine calls a number of subroutines: SETBC, KNOBS, MCALL2D, IMPEX, PCALL2D, TSTEP and BALANCE. In addition, the subroutine SETPROP is called several times to determine the properties. The first subroutine, SETBC, sets boundary conditions for the fluid dynamics calculation. SETBC extracts the boundary conditions for the current time step from tables of the boundary velocities, void fractions, and temperatures versus time. These tables are specified in the input file (see Appendix B).

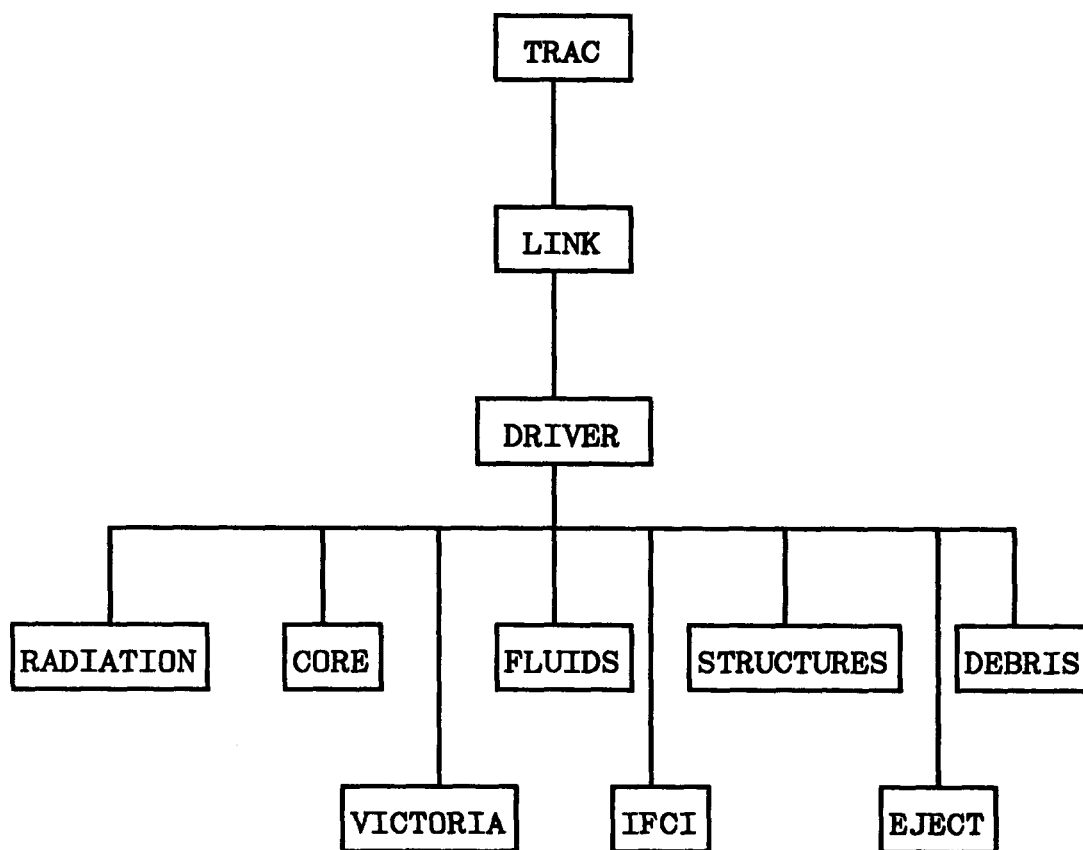


Fig. A.1 Overall MELPROG Structure.

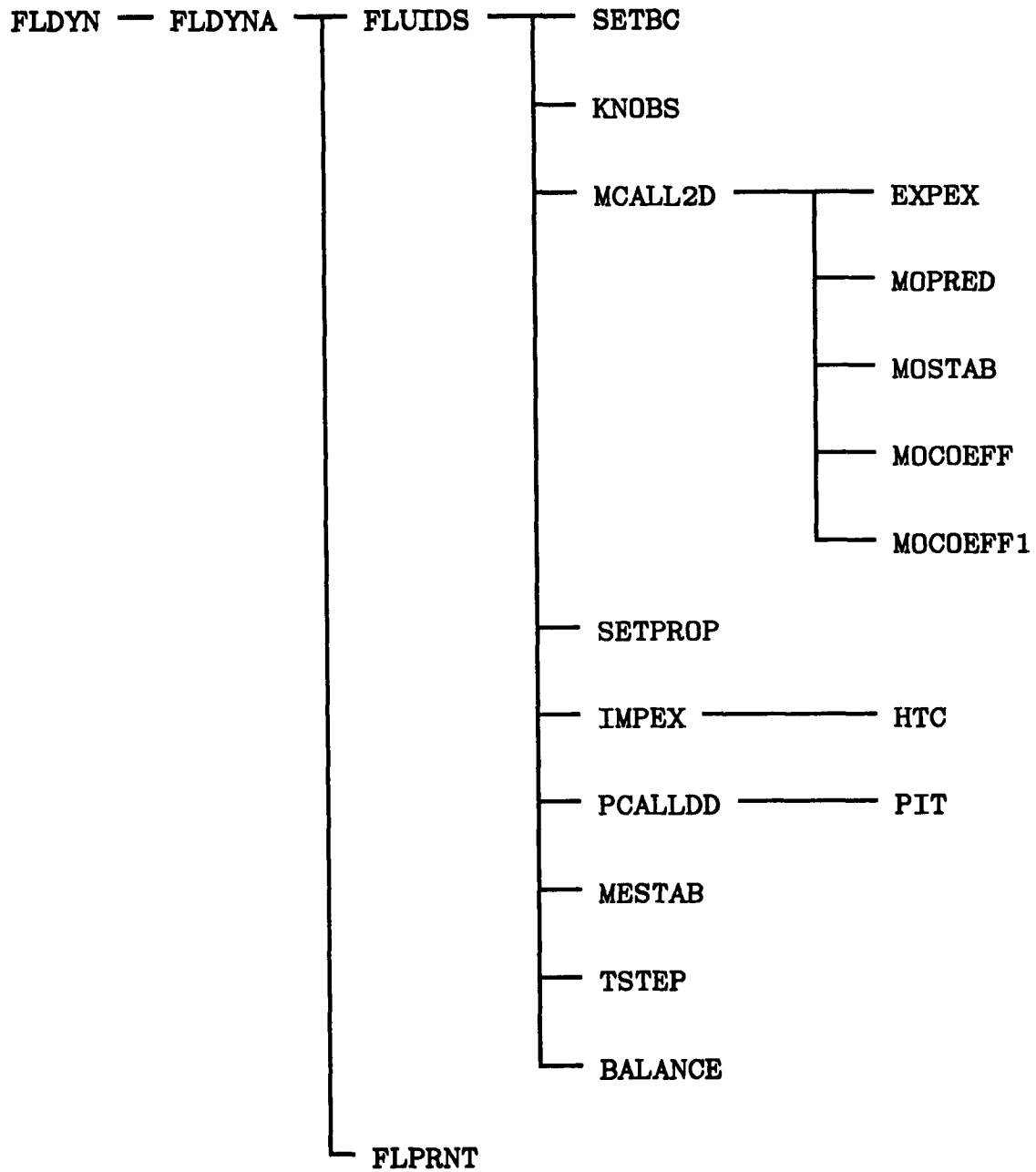


Fig. A.2 Functional Diagram of FLUIDS Module Routines.

The KNOBS subroutine is a control routine for the fluid dynamics equation solver. It selects the fields in each mesh cell that are not present (void fraction below a small value, e.g., AL10) and also do not have a source term (mass transfer, convective transport) that would increase the void fraction. These fields have their corresponding mass, momentum and energy equations turned off within the affected cell.

The next subroutine, MCALL2D, is a driver that sets up the matrix for the solution of the momentum equations. Friction factors for each fluid field are calculated explicitly in the EXPEX subroutine. The initial "predictor" step for the semi-implicit Stability Enhancing Two-Step (SETS) solution technique is conducted in the MOPRED subroutine. MOPRED uses old values in the convective terms of the momentum equations to decouple them spatially: these equations are then solved simultaneously on a cell-by-cell basis, treating the interfield momentum transfer terms implicitly. MOSTAB uses the predicted velocities in the interfield terms to decouple the momentum equations between fields: these equations are then solved on a field-by-field basis, this time treating the convective terms between cells implicitly. This produces "stabilized" velocities.

The MOCOEF and MOCOEF1 subroutines are then used to solve the momentum equations for velocity as a function of pressure gradient explicitly. The stabilizer velocities are used in the interfield terms to decouple the fields. This step results in the velocities being expressed in the form:

$$v_{k+1/2}^{n+1} = f(p_k^{n+1}, p_{k+1}^{n+1}) \quad . \quad (A.1)$$

These expressions are then substituted directly into the mass and energy equations for the pressure iteration.

The pressure iteration begins by calling SETPROP, this time using new properties. Heat transfer coefficients, mass transfers and heat transfer rates are calculated in IMPEX. The actual Newton-Raphson pressure solution is performed by the JAC41H routine which is called by PIT twice during each iteration. On the first call JAC41H sets up the matrix elements for the Jacobian and the second time it back substitutes to determine the solution. The PIT subroutine returns a value in ICONV of either 1 (continue), 2 (converged) or 3 (convergence failure).

At this point, FLUIDS either continues the pressure iteration by looping back to JAC41H (ICONV = 1), exits back to the main driver program to have the time step reduced in TSTEP (ICONV = 3), or exits the pressure iteration loop (ICONV = 2). In the latter case, SETBC, SETPROP and IMPEX are then called to update boundary conditions, fluids properties and transfer terms to the new time level, using the new fluids values. The MESTAB subroutine is called to perform the final stabilizing solution of the mass and energy equations. This ends the SETS solution of the fluids equations.

The BALANCE subroutine evaluates the mass and energy residuals. This is done by calculating a mass and energy balance for the solution using the first two fields and the energy sources for the other modules.

The final routine, FLPRNT, outputs FLUIDS module quantities on the print file whenever the print flag is set (at EDINT internals, or at the problem end time).

A.2 CORE Module

This section describes the functional layout, purpose, and operation of the CORE module. The driver for this module calls a number of subroutines (see Fig. A.3). The first subroutine, COREHT, performs the conduction heat transfer calculation. The CORE module can use many axial cells within each FLUIDS cell. Within every axial CORE cell, one-dimensional radial heat conduction is considered for every representative core structure. Heat transfer geometry coefficients are calculated in the CRGEOM subroutine. The CORPOW subroutine determines power generation due to decay heating, oxidation of the Zr cladding and U-Zr-O eutectics (OXZRUO) and oxidation of stainless steel (OXSTEL). This subroutine will be modified at a later date to account for the oxidation of B_4C in boiling water reactors. Boundary conditions are then set in the COREBC routine; radiation and convection from structure surfaces are considered. The COREHT routine then solves the conduction equations using a modified Crank-Nicholson algorithm.

When molten materials are present, candling is treated in the CANDLE subroutine. CANDLE allows materials to relocate on the outside of core structures, inside of gaps (e.g., the gap

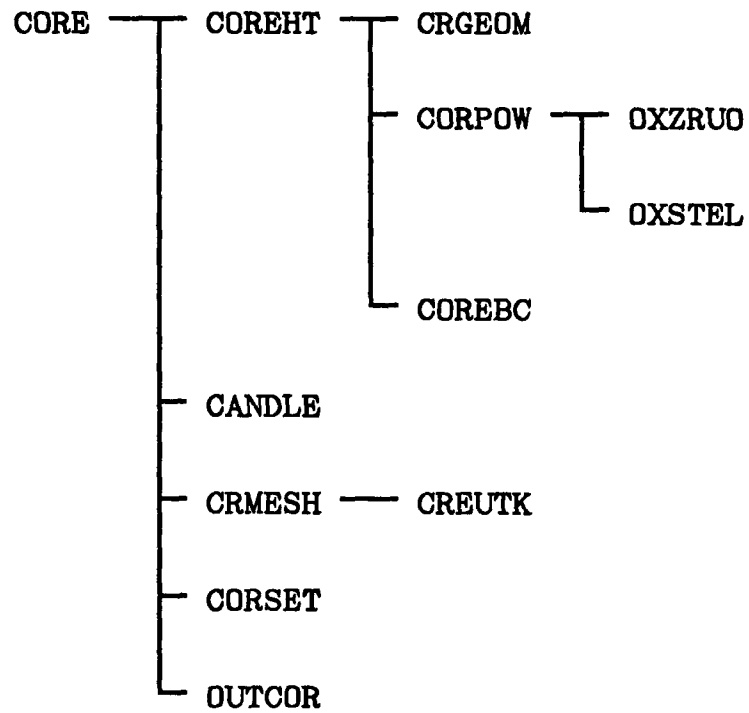


Fig. A.3 Functional Diagram of CORE Module Routines.

between the fuel rod and the clad) and through holes and cracks that can form in the clad. Currently, CANDLE only considers film flow. A one-dimensional momentum equation is solved along the flow path; the model accounts for the convection of heat downward by the melt as it flows downward.

The CRMESH routine sets up the radial mesh and determines material regions for the heat transfer calculation at the next time step. New regions can be formed for a variety of reasons. For example, when a significant fraction of the clad is oxidized, two regions are created: a ZrO_2 region on the boundary and a α -Zr zone between the Zr and the ZrO_2 . New regions are also created when a eutectic reaction occurs (these reactions are treated in the CREUTK routine) and when molten materials flow downward into a new axial CORE cell. In addition to creating new regions, CRMESH determines if two adjacent regions can be combined and if a material region should be deleted.

The next subroutine, CORSET, calculates information for the other modules. For example, core structure volume fractions, flow area factors and hydraulic diameters are updated for use in the fluid dynamics calculation. In addition, an average temperature is calculated in each FLUIDS cell for the radiation heat transfer calculation.

The last subroutine, OUTCOR, prints CORE module information in the output file whenever the print flag is set (at EDINT intervals and at the problem end time).

A.3 STRUCTURES Module

This section discusses the STRUCTURES module which controls the thermal and mechanical calculations for structures. The routines in this module are divided into two functional sets: heat transfer routines and mechanics routines. Each of these sets is discussed separately in the following sections.

Subroutine STRUCT is the driver routine for the STRUCTURES module. STRUCT is divided into five parts. Part 1 performs data transfer from other modules to the STRUCTURES module. Part 2 consists of calling subroutine STRCHT to perform structure heat transfer calculations. Part 3 performs the structure mechanical calculations. Part 4 calls subroutines STGEOM and STRSET to calculate changes in structure geometry and fluid flow areas and hydraulic diameters, respectively. Part 5

calls subroutine OUTSTR to print structures information.

A.3.1 Heat Transfer Routines

This section describes the routines which perform the structure thermal analysis. The calling sequence for structure heat transfer calculations is shown in Figure A.4.

Subroutine STRCHT performs the heat transfer solution for each structure. This routine calls subroutine BCSSET for boundary conditions on each structure surface. STRCHT sets up the heat transfer matrix which is then solved by subroutine TRYSLV. After solving for the temperature distribution, it checks for melting and performs an iteration on the specific heat if melting is occurring. If a structure is melting, subroutine REMESH is called to calculate changes in the structure geometry and to determine if failure by complete melting has occurred. Finally, STRCHT calculates the net heat transfer from the structures in a cell to each of the fluid components in that cell.

Subroutine MELPRO (see Chapter 10) calculates thermal properties for use in the structure heat transfer calculation. MELPRO is called by subroutines STRCHT and STGEOM.

Subroutine BCSSET calculates thermal boundary conditions for heat transfer between structures and fluid components. BCSSET treats one boundary condition on surface 1 and three boundary conditions on surface 2 of each structure. BCSSET is called from subroutine STRCHT.

Subroutine TRYSLV is a matrix solution routine for non-symmetric tridiagonal matrices. TRYSLV does not destroy the original matrix and it is called from subroutine STRCHT.

Subroutine REMESH is used to calculate changes in a structures geometry that result from melting. REMESH determines mass transfer for structural material to the fluids. It recalculates the nodal heat transfer mesh in a structure that is melting and it determines structural failure by complete melting. REMESH is called from subroutine STRCHT.

Subroutine STGEOM calculates the initial nodal heat transfer mesh for each structure. In addition, it calculates heat transfer geometric factors and nodal masses for the initialization and whenever the nodal heat transfer mesh is

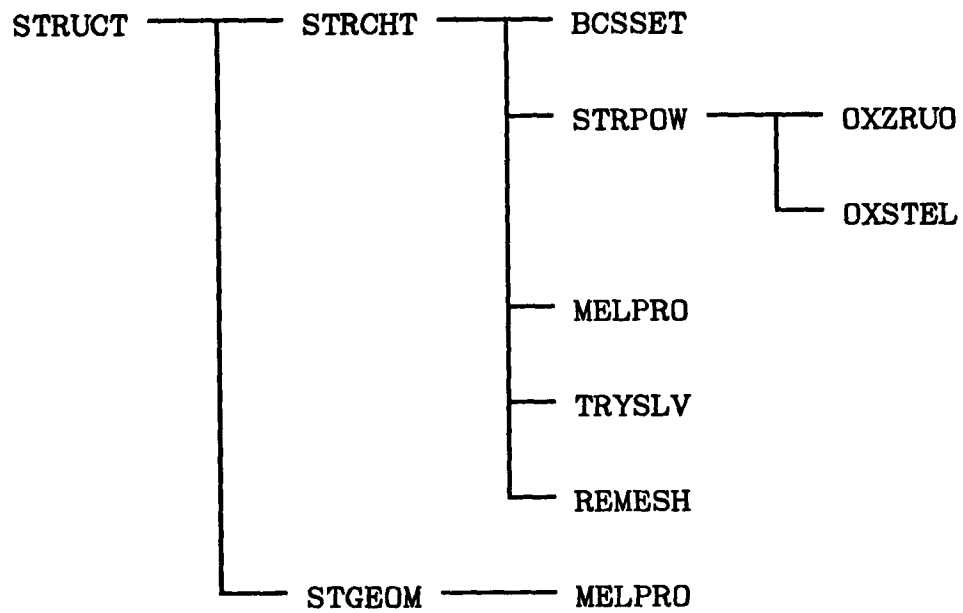


Fig. A.4 Functional Diagram of Structure Heat Transfer Routines.

changed, for example by melting. STGEOM is called from subroutine MPRGIN during initialization and from subroutine STRUCT at each time step. STGEOM calls subroutine MELPRO to get material densities.

A.3.2 Mechanics Routines

This section describes the functional layout, purpose and operation of the mechanics routines in STRUCT. The routines are set up functionally as shown in Figure A.5.

Unfortunately, the FLUIDS module is unable to handle an instantaneous change in the structure volume fraction. Also, there are some restrictions on the number and position of structures within a cell in MELPROG so that the continuation of the heat transfer calculation for a failed structure following relocation could cause problems in the STRUCT module. Structure relocation has been restricted to occur only by transfer into the melt field of FLUIDS by a mass transfer function following melting, pending resolution of these problems with the solid lump relocation. The loading diagram, however, is changed to reflect any new orientation caused by failed structures.

DEBID is the first step in determining the loads on the structures. This routine is responsible for determining where debris is located and which structure is supporting the debris, if any. The ID numbers of the supporting structures are stored in an array, LSDBAS. The routine also assigns unique identifying numbers to each separate debris bed, taking into account possible dividing walls and plates. The ID numbers are assigned in ascending order, except that, if debris is connected to debris in the cell immediately below, the ID is the same as that for the debris below, and, similarly, debris connected to debris in the next inner ring also assumes the ID of that debris.

The next routine, LOADZ, sets up the loads for the structures. This is done in three passes. The first pass forms cell loads on each structure from hydraulic forces, the weight of the part of the structure in that cell, fuel rods being directly supported, or debris (using the support ID numbers calculated in DEBID). Loading forces are determined for both the radial and axial directions. The structures for which loads are found include any failed pieces of structure, which are

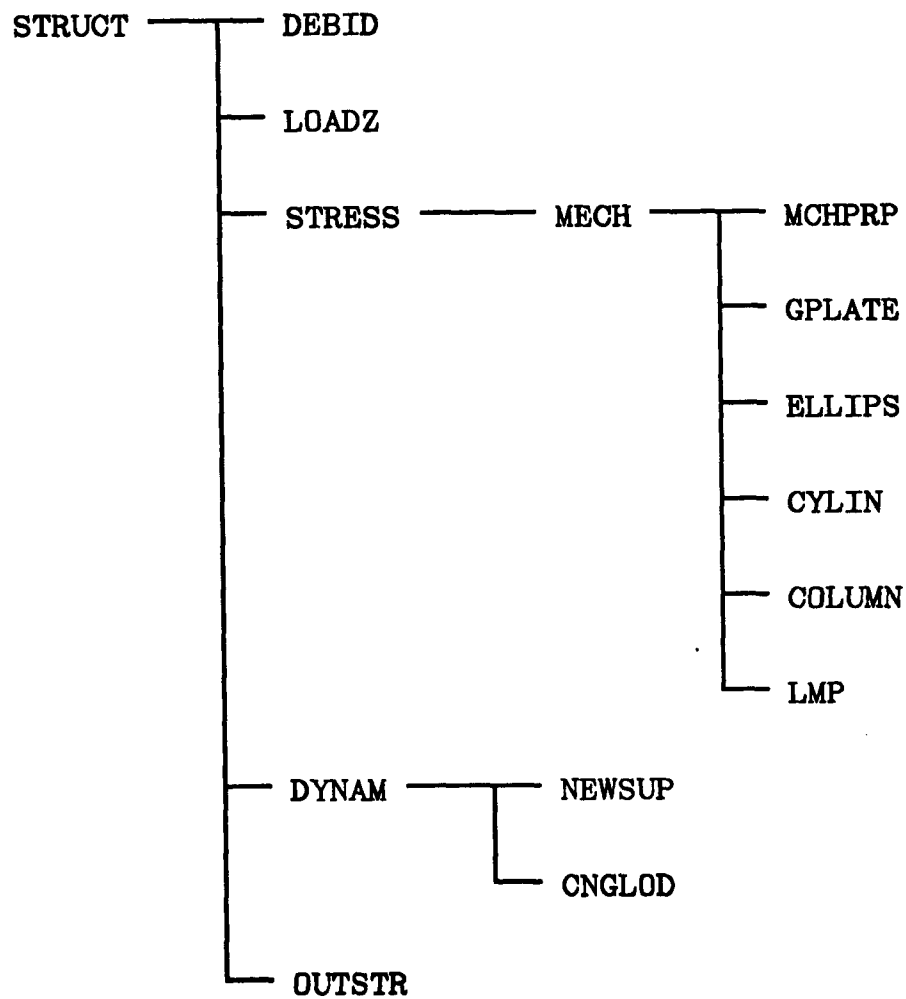


Fig. A.5 Functional Diagram of Structure Mechanics Routines.

termed "lumps" and identified by unique ID numbers greater than NST+1.

Pass 2 uses the cell loads found in pass 1 to initialize the ring table. This table contains a row for each intact or failed structure and has NSR rows. The row contains the total forces in the radial and axial directions as well as the cell specific forces for each cell containing part of the structure. These forces will include the contributions from supported structures and all parts of the supporting structure at the end of pass 3; pass 2, however, initializes the ring table to only the self-load for each structure. The data storage arrangement is shown in Table A.1. Each row (structure) has entries for the total radial and axial forces in the first two columns. The following entries are either total forces in an axial node for axial structures (e.g., walls or columns) or total forces in a radial ring (plates).

Table A.1				
Ring Load Table Data Structure				
IDCEL:	F_r tot	F_a tot	F_r 1 F_a 1...	F_r NF F_a N
1:				
2:				
.				
.				
.				
NSR				

Pass 3 completes the ring table setup by adding in the forces from the supported structures, using the load table to determine which structures to add. Information about the type of structure is also used to determine the appropriate way to apply the loads.

The STRESS routine uses the ring table to get the loads in a cell for each structure. These loads are passed, along with the structure average temperature, to MECH, which is a control routine for the mechanics models. MECH calls MCHPRP to get the mechanical properties at the average structure temperature and then calls the appropriate mechanical model to evaluate the stresses. MECH then calls LMP, The Larson-Miller Parameter routine, to find the predicted rupture time and update the life fraction. If the life fraction exceeds 1.0, indicating that the

rupture lifetime has been used up, the structure failure flag is set and passed back to STRESS. There is a separate failure flag for the structure in each cell containing the structure.

STRESS calls UPFAIL after MECH if the failure flag in a cell has been set. UPFAIL sets the failure flags for the structure in connected cells, if those parts of the structure were supported by the failed section of structure. For instance, if a wall supported from above fails in a node, UPFAIL sets the failure flags for the sections of wall below the failed node.

The next main routine, DYNAM, is responsible for the dynamics of moving failed pieces of structure from cell to cell. Currently, DYNAM is a very simple model that assumes all structure pieces eventually end up falling under gravity. The model checks the cells below a failed piece of structure until it encounters an intact structure capable of supporting the failed piece. (A support structure is defined to be a plate or column.) This structure will receive the load from the failed piece in subsequent time steps. If there is no intervening obstruction, such as a previously failed structure (now called a lump), debris, or fuel rods, then the failed piece is set up to be moved to the cell of the new support structure (or to the cell above it, in the case of the support being a top plate) on the next time step. This phase of moving the new lump is done in NEWSUP, called from DYNAM. In the event that no new support is found, the lump will be moved to the bottom cell in the problem. If the support is in the same cell as the lump, no move is performed, but the load is transferred. The move from the old cell to the new is set up by storing pointers to the failed-piece data and to the new cell in the LSLUMP array.

Intervening debris or fuel rods are not classed as supports because the debris or fuel rods are assumed to pass their load onto whatever is supporting them, for example, a plate. Therefore, for purposes of switching the load pattern, the load of a lump is transferred downward to the first support structure encountered.

The mechanism for changing the loading pattern is to switch the support pointer for the failed structure in the load table to point to the new support. The failed structure is then assigned a "lump" ID number (a structure ID number greater than NST+1). Any structure previously supported by the failed piece will now be pointing to this lump, so the load table is searched

for structures pointing to the now-failed structure, and their pointers switched to point to the new support. The load chain switching is performed in CNGLOD.

The load patterns are set up initially in routine LINK. The load patterns are partially determined by the type of structure, the type of mechanical model, and the relative locations of the structure and its support, according to assumptions about how certain structures are supported. These assumptions are as follows:

1. An in-cell support is unconditionally accepted.
2. If a support is in the next cell below, it cannot be a bottom plate.
3. If the structure is a column and is also in the cell below, the structure will be assumed to be supported by itself in the cell below.
4. If a support is in the cell above, it cannot be a top plate.
5. Walls and lumped parameter structures will be assumed to be supported by themselves from the cell above, if they are in the cell above.
6. If a support is in the next cell out, it cannot be an outer wall.
7. Elliptic plates and eggcrate grills are supported by themselves in the next cell out, if they are in the next cell out.

Structures not satisfying any of the above conditions will cause LINK to fail, and MELPROG will exit on an error after printing the loading diagram and appropriate error messages. Note that walls are supported from above, columns from below, lumped parameter structures from above, elliptic and eggcrate plates from the outer edge, and grid plates from a structure in the same ring (i.e., either sitting on columns or hanging from lumped parameter structures). Also, supporting structure for grid plates must exist in each ring containing the grid plate.

At the end of LINK, the load chains describe the complete

loading pattern in the problem. For example, the core barrel supports the core support plate, which in turn holds up the support columns, which support the lower core grid plate, etc. An identification number greater than or equal to 900 is used in the input to indicate an external support for a structure, as for instance the support for the reactor vessel. These artificial "pin points" are entered directly into the load table as their negatives (e.g., -900), rather than having pointers to their locations in the table.

A.3.3 Flow Areas and Hydraulic Diameters

Subroutine STRSET dynamically calculates fluid flow areas and hydraulic diameters for use throughout MELPROG. These calculations are performed here rather than in the FLUIDS module because the STRUCTURES module contains the detailed geometric information needed to perform the calculations.

A.4 DEBRIS Module

This section discusses the main subroutines in the DEBRIS module. Subroutine DEBRIS controls the DEBRIS module calculation. This routine calls the other routines which perform the initialization and analysis of the debris beds. Figure A.6 illustrates the calling sequence for the major subroutines in this module.

At the start of the calculation, the DEBRIS subroutine determines the corium volume fraction distribution on a two-dimensional basis. Most of the information is obtained from FLUIDS arrays. With this information and the approximations discussed in Chapter 6, the local corium volume fractions can be calculated.

The first routine, DBFORM, decides whether or not a debris bed is forming based on an empirical blocking correlation. If a debris bed forms, then the next routine, SEARCH, determines whether dryout has occurred. If the bed is dry, then subroutine FINDIC is called to determine the initial conditions. After a bed has formed, subroutine BEDBC is called to determine the boundary conditions.

When debris beds exist, subroutine DBSRC prepares the arrays which transfer mass and energy to FLUIDS. The first call to this routine initializes the arrays and the second call actually

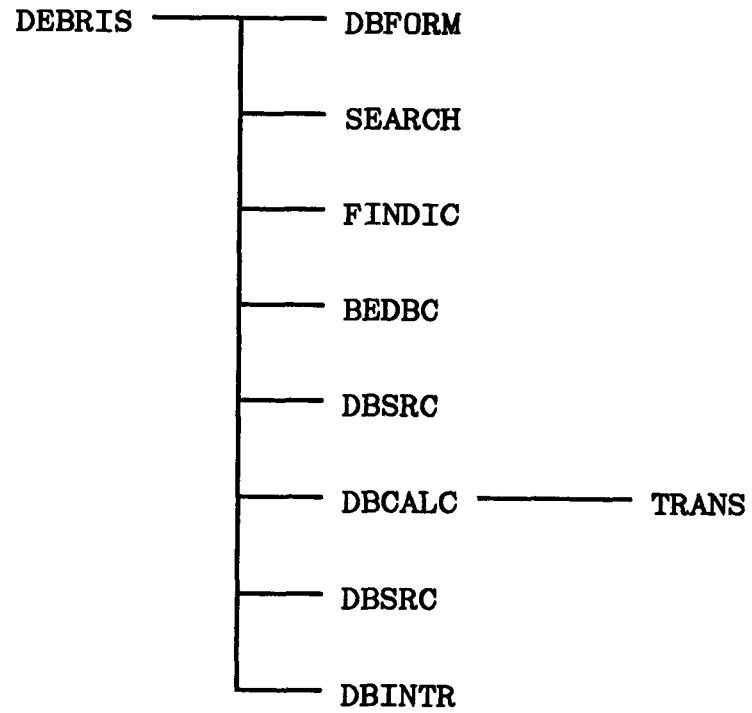


Figure A.6 Functional Diagram of DEBRIS Module Routines.

sets the values.

Subroutine DBCALC controls the debris bed calculation. This routine sweeps through all debris beds calling subroutine TRANS for each one. TRANS solves the detailed heat transfer and material relocation equations.

After solving the equations at a given time step, subroutine DBINTR is called to obtain information to be transferred to other modules.

A.5 RADIATION Module

The RADIATION module consists of two main subroutines, RADMIM and VIEWFC. There are four additional subroutines: an input and initialization subroutine called RADIN; two pointer calculation subroutines called RADPNT and RDPNTT; and a subroutine to calculate the Planck mean absorption cross section for steam called FKPM. Subroutines RADIN and RDPNTT are called from the INPUT module. RADMIM is the driver subroutine for the RADIATION module. Figure A.7 shows the calling sequence for the RADIATION module.

Subroutine RADMIM is divided into six sections. There are four parts to the calculation sequence, a data transfer section and an output section. The data transfer section transfers needed information from each of the other calculational modules to the RADIATION module. The first part of the calculation sequence includes calculation of absorption coefficients, emissivities, black body emission spectra. Subroutine VIEWFC is called to calculate view factors and mean transmission coefficients. In the second part, a coefficient matrix and part of the resultant vector are assembled for each cell. Also, the coefficient matrices are inverted. In the third part, an iteration on the cell boundary fluxes is performed. The changing parts of the resultant vector are calculated and the total flux leaving each surface is calculated. The fourth part calculates the net radiation fluxes to each surface and to each fluid component. The final section prints radiation heat fluxes and stores permanent radiation data. RADMIM calls subroutines FKPM, VIEWFC, SGECO, and SGESL. Subroutines SGECO and SGESL are system utility routines used in the matrix solution.

Subroutine VIEWFC calculates view factors and mean transmission coefficients. View factors are calculated the

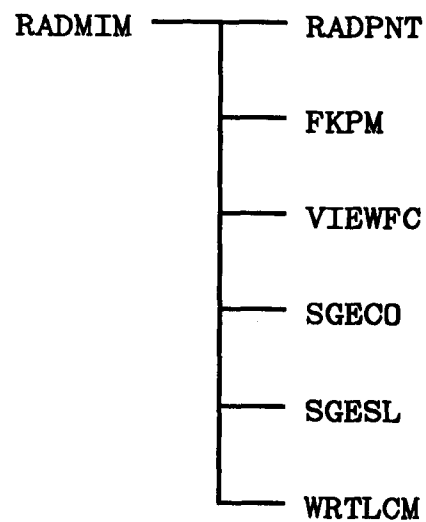


Fig. A.7 Functional Diagram of RADIATION Module Subroutines.

first time that VIEWFC is called and whenever the geometry of a cell changes. Mean transmission coefficients are calculated each time VIEWFC is called. VIEWFC is called from RADMIM once per time step. VIEWFC calls no subroutines.

A.6 TRAC/MELPROG Link Computational Flow

Solution of the fluid dynamics equations by means of the SETS method requires at least six passes through each one-dimensional TRAC component for every time step. The first two passes are made in the prepass step to solve the stabilizer motion equations. This step first obtains the velocities interior to a component as a function of the junction velocities. When the reactor vessel is being handled by MELPROG, this function is accomplished by the subroutine DCORE1 that in turn uses the section of FLUIDS that calls MOPRED and MOSTAB. Next, the motion equations are applied and a closed set of equations is obtained for the junction velocities. This system is then solved for the junction velocities. The interior velocities are then obtained by a backward substitution using the known junction velocities. For a MELPROG-treated core, this is handled by a call to DCORE1 that in turn calls DCORE1A. The next two or more passes are to solve the basic hydrodynamics equations. The flow for the so-called outer iteration is shown in Figure A.8. A single complete pass through this procedure provides the solution for the linearized finite-difference equations. Subsequent passes through the procedure for the same time step produce a Newton iteration on the non-linear difference equations.

The final two passes made in the postpass are to solve the stabilizing mass and energy equations. The solution of these equations is quite similar to that of the stabilizing motion equations and is handled for a MELPROG core simulation by the section of FLUIDS that calls MESTAB on the first pass. The back substitution pass to complete the solution is handled by DCORE3D. Before returning to TRAC, the final MELPROG fluids pass is followed by calls to the MELPROG routines that perform the rod, radiation, debris bed, and structures calculations.

A.7 MELPROG Data Input and Initialization Module (INPUT)

MELPROG is a modular code in which the data input and initialization are also modular. Data input and initialization for all modules is controlled by subroutine MPRGIN; however, specific input for each calculational module is through its own

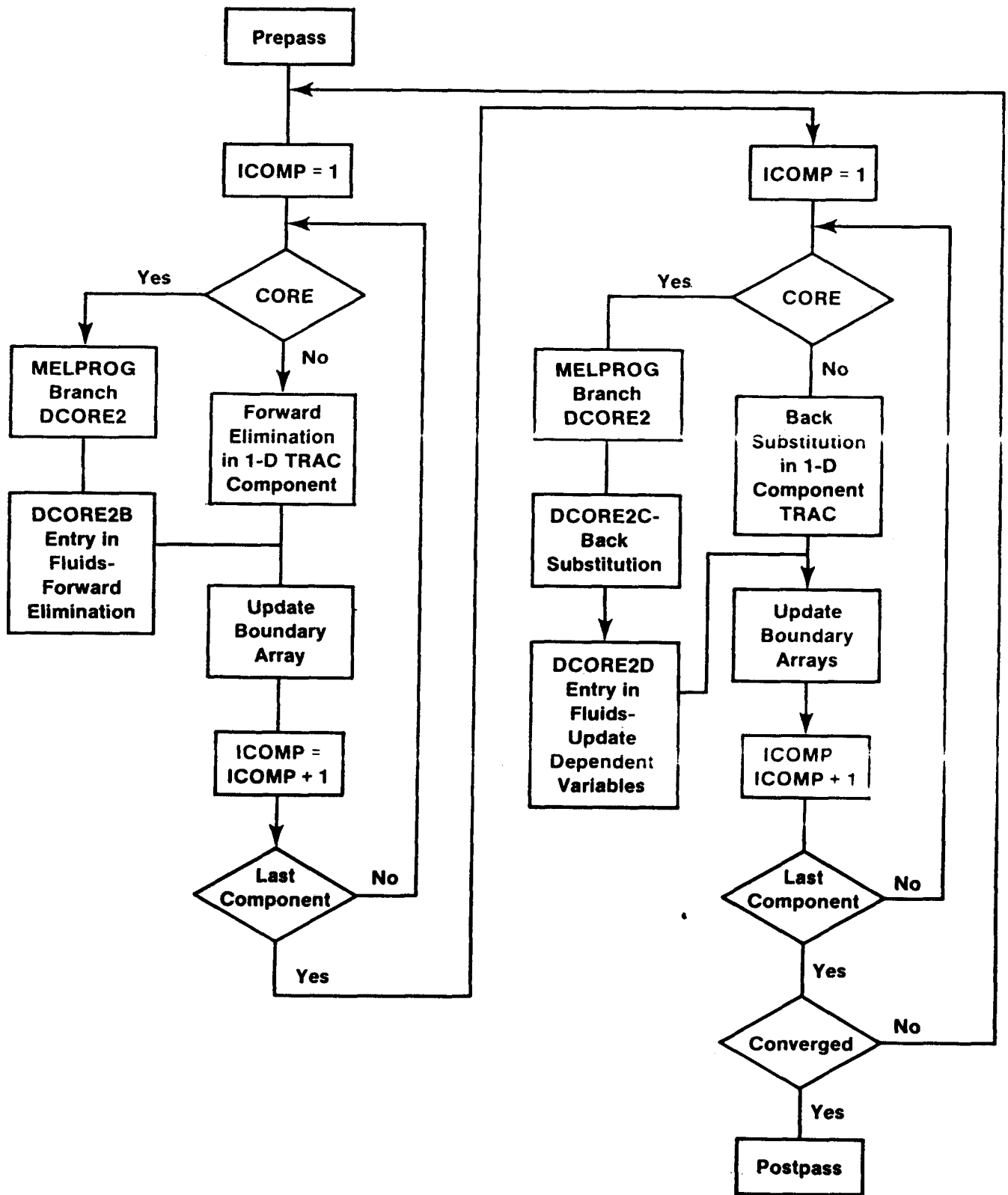


Fig. A.8 Outer iteration flow chart for TRAC/MELPROG.

input subroutine. Similarly, initialization for each calculational module is through one or more subroutines written specifically as part of that module. The calling sequence for the major subroutines that are part of the INPUT module is shown in Fig. A.9. A brief description of subroutine MPRGIN and the other input routines for each module are given in this section.

In addition to acting as the driver routine for the INPUT module, subroutine MPRGIN serves as the input routine for the "general input." MPRGIN calls several subroutines to read and initialize the input for the various calculational modules. As part of the input processing, MPRGIN checks for sufficient storage space in the X array for each module. (Data storage in the X array is discussed in Section A.8.) If any module is found to require more space than is available with the existing array dimensions, an error message is printed which indicates the amount of storage space available and the amount required. If it is found that insufficient space exist for any module, the calculation is stopped after completion of the input processing.

Subroutine FLDIN is the data input routine for the FLUIDS module input. FLDIN reads FLUIDS scalar and array data and also reads FLUIDS boundary conditions. FLUIDS initialization is performed after all input has been read for the other modules. (Fluid volume fractions, flow areas and hydraulic diameters can be initialized for the fluid dynamics calculation after reading CORE and STRUCTURES data.) FLDIN calls two subroutines: SETPROP and SETBC to initialize the fluids properties and the boundary conditions, respectively.

Subroutine COREIN is called to read the CORE module input. The COREIN subroutine subdivides each FLUIDS cell in the axial direction. The MPRGIN routine then calls the CRINIT subroutine that sets the power factors and determines the initial temperature and surface area of each structure. This subroutine also sets volume fractions, flow areas and hydraulic diameters for the FLUIDS module.

Subroutine STRCIN is the main input routine for STRUCTURES module input, heat transfer and mechanical data. Mechanical property data are read by subroutine MECHIN. In addition to reading data STRCIN calculates array space requirements for the structure heat transfer and mechanical data arrays. The array lengths are used in the calculation of pointers to locations in the X array (see Section A.8). Initialization of data for the structure heat transfer is performed in subroutine STGEOM (see

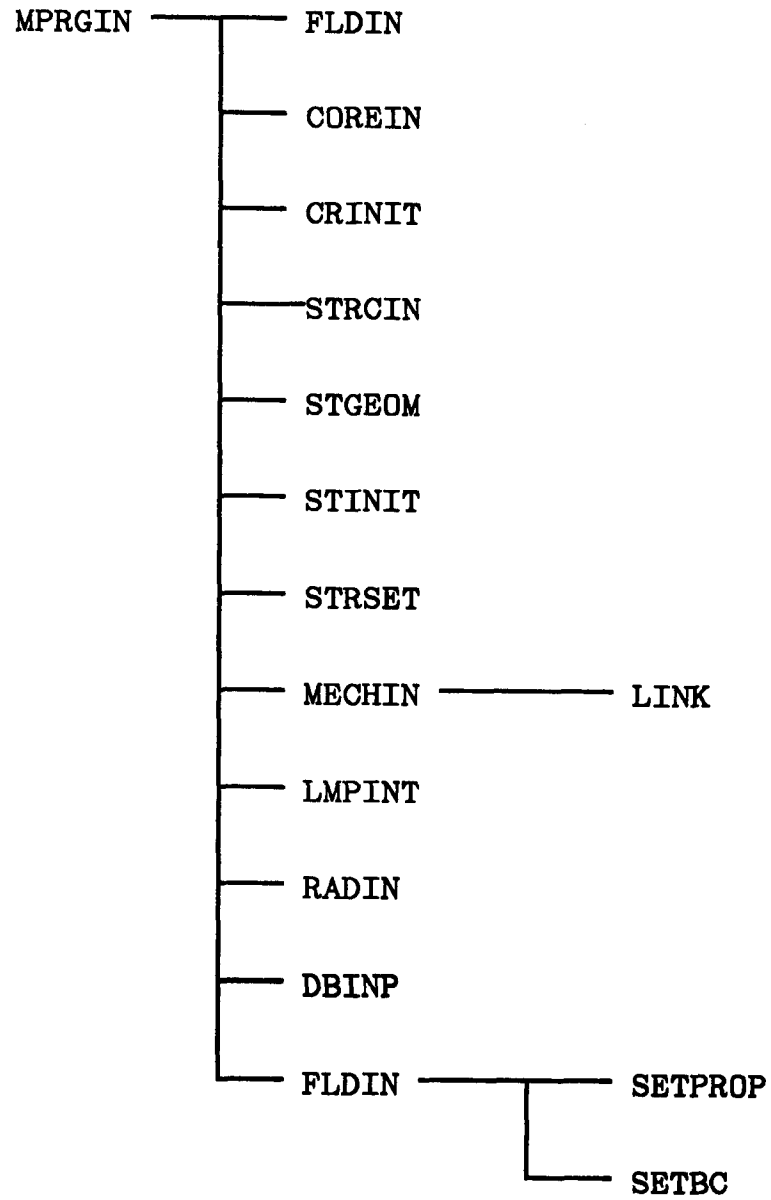


Fig. A.9 Functional Diagram of INPUT Module Subroutines.

Section A.3). Subroutine STRSET is called as part of STRUCTURES module initialization to calculate flow areas and hydraulic diameters (see Section A.3) for use in the FLUIDS module initialization.

Several subroutines are used to initialize the structure mechanics data. The first subroutine in this initialization is STINIT. STINIT processes the input problem description to set up a table which specifies the geometry for all structures. The geometry table (Table A.2) contains a row for each distinct structure. The rows are accessed by the structure identification numbers IDCEL. There are NST of these identification numbers, one for each structure, which are assigned in the input for the problem. Each row of the geometry table contains the mechanical type number, the mechanical model number, and geometric information describing the structure for that row.

Table A.2				
Geometry Table Data Structure				
IDCEL:	Type	Model	Material	Geometry Data
1:				
2:				
.				
.				
.				
NST:				

The next initialization routine, STSET, clears the loading table, which specifies the loading pattern for each structure, and calls LINK, which sets up the table. The loading table is constructed from the information given in the input via IDCEL (the structure ID number) and IDSUP (the support structure ID number), plus information on the type of structure and its location in the reactor vessel from the geometry table. The loading table (Table A.3) contains a row for each problem cell. Each row contains three entries for each structure in the cell. These three entries consist of (1) the structure ID number IDCEL, (2) a pointer to the location of the supporting structure in the load table, and (3) a flag that is used to prevent circular loops from occurring due to incorrect user input. LINK forms the load chains in two passes: in the first pass, the

structure ID numbers are inserted into the load table; in the second pass, the pointers to the supporting structures are determined.

Table A.3	
Loading Table Data Structure	
CELL	((ID 1) (Support Pointer) (Flag)) ...
1	
2	
.	
.	
.	
NCELL	

MECHIN reads in the mechanical property tables for NMATMX materials from an alternate input file (MELMCH). These properties are the elastic modulus, yield strength, ultimate tensile strength, Poisson's ratio, and mean coefficient of thermal expansion as functions of temperature. Following these tables, the rupture time table is read in as a function of stress and temperature. It is necessary to have at least one rupture time for each stress point entered. The rest of the times for the given stress (at the other temperature points) can be zero. The unknown times are then filled in by the LMPINT routine using interpolation or extrapolation as appropriate. This completes initialization of the mechanics routines.

Subroutine RADIN reads the RADIATION module input data. It also calculates array space requirements and a special pointer array for location of two-dimensional cell radiation boundary fluxes. RADIN is called by MPRGIN.

Subroutine DBINP is used to read the DEBRIS module input data. DBINP also initializes a number of constants and flags used throughout the DEBRIS module.

A.8 Data Management Structure Used in MELPROG

Data storage in MELPROG is divided into two categories, permanent and temporary. Permanent data are data that must be kept from time step to time step or that need to be passed from one module to another. Temporary data are data that are transferred into a module from another module and data needed

during a time step but not from one time step to another (working storage). All permanent data arrays, referenced by pointers, are stored in a single large array called XLCM. Access to data stored in XLCM is by blocks that correspond to each of the calculation modules. The permanent data are transferred to the working array X for use in a given module. All calculations using array storage, with a few exceptions, are performed in the X array. No direct access to the XLCM array is allowed. On a CDC 7600 computer, the XLCM array is stored in Large Core Memory (LCM) while on a CRAY computer, or any single memory machine, it is stored in main memory. The X array is stored in Small Core Memory (SCM) on a CDC 7600 and in main memory on CRAY or other single memory computers.

Subroutine RWLCM is used to perform the transfer from the XLCM array to the X array and from the X array to the XLCM array. Three entry points (RWLCM, REDLCM, and WRTLCM) are used. RWLCM is used for initialization of the storage in the XLCM array. REDLCM is used for reading data from the XLCM array to the X array. WRTLCM is used to write data from the X array to the XLCM array. All data for a particular module is transferred to or from the XLCM array in a single block. Currently, the data transfer is necessary for both single memory and multi-level memory computers; however, in future versions of MELPROG the data management will be changed for single memory computers. The actual cost of the data transfer on a CRAY is approximately 1% of total computer cpu time.

All of the entry points (RWLCM, REDLCM, and WRTLCM) use the same argument list as demonstrated here by a call to RWLCM:

```
CALL RWLCM(NBLOCK,IRING,IXADDR,LENGTH)
```

where

NBLOCK = internal I/O unit number, IOFLDS for the
FLUIDS module, IOPINS for the CORE
module, IOSTRC for the STRUCTURES
module, IORAD for the RADIATION module,
and IOBED for the DEBRIS module,

IRING = 1 for all modules,

IXADDR = beginning address in the X array to transfer data to or from, and

LENGTH = number of words to be transferred to or from the X array. For data transferred to the X array, LENGTH is returned, while for transfers from the X array to the XLCM array, LENGTH must be supplied.

When data is transferred to the XLCM array from the X array, the value of LENGTH must not exceed the value used to initialize the XLCM array. If a call is made to entry point WRTLCM with a value of LENGTH greater than the value used to initialize the storage, the code will stop on an error.

The number of modules for which data must be transferred is dependent on each module. The FLUIDS and RADIATION modules interact with all the other modules and, therefore, must get information from each of them. The CORE module interacts with the FLUIDS and RADIATION modules. The STRUCTURES module interacts with the FLUIDS, RADIATION, and DEBRIS modules, while the DEBRIS module interacts with the FLUIDS, CORE, RADIATION, and STRUCTURES modules.

Data storage in the X array is divided into three groups. The groups are permanent data, transfer data, and working space. In the FLUIDS, CORE, and RADIATION modules, the permanent data is stored first. The transfer data is stored next and the working space storage is at the end. In the STRUCTURES and DEBRIS modules, the order of storage is transfer data, permanent data, and working space.

B.0 INPUT INSTRUCTIONS

In the next section, the input instructions required by the MELPROG/PWR-MOD1 code to start a calculation are briefly described. A detailed description of the input and output will be provided in a user's manual that will be published in the near future. General data, such as the times at which data should be printed and the level of detail of the output, is input first. FLUIDS module data is input next: for example, information on the fluids grid, friction factors, boundary conditions and initial conditions. Information for the CORE module is then provided (descriptions of the models for the fuel rods, control rods, etc., are given in terms of material regions). General structures are then defined in terms of geometry and boundary conditions. Only a small amount of input is required for the RADIATION and DEBRIS modules: these modules get most of the data they require from other modules.

B.1 MOD1 Input Instructions

B.1.1 General Input

1. ITITLE (20A4)
 - A. ITITLE = problem title (up to 80 characters).
2. IRESTRT
 - A. IRESTRT = the restart switch.
3. DMPINT
 - A. DMPINT = restart dump interval (sec) (on MELDMP).
4. GFINT
 - A. GFINT = graphics dump interval (sec) (on CMOUT).
5. EDINT
 - A. EDINT = full edit interval (sec) (on MELOUT).
6. SEDINT
 - A. SEDINT = short edit interval (sec) (not used).
7. PRNTTO
 - A. PRNTTO = problem time or step number at which a full edit is desired.
8. IPRTF, IP RTP, IP RTR, IP RTS, IP RTD
 - A. IPRTF = fluids full print flag.
 - B. IP RTP = core full print flag.
 - C. IP RTR = radiation full print flag.
 - D. IP RTS = structures full print flag.
 - E. IP RTD = debris full print flag.
9. KMAX
 - A. KMAX = number of axial nodes.

10. NRING
A. NRING = number of radial rings.
11. NCORE1
A. NCORE1 = axial node number for core bottom.
12. NCORE2
A. NCORE2 = axial node number for core top.
13. NBOTM
A. NBOTM = axial node number containing vessel bottom.
14. ICORER
A. ICORER = radial ring number for extent of core.
15. LRAD
A. LRAD = radiation switch.
16. GASCOEF
A. GASCOEF = maximum fraction of gas internal energy that the gas can receive from all heat sources in one time step.
17. TIME
A. TIME = the starting time (sec).
18. ENDTIM
A. ENDTIM = problem end time (sec).
19. DELTAT
A. DELTAT = the initial time step (sec).
20. NTIM
A. NTIM = number of time step pairs in maximum allowable time step table (>1).
21. ((STEP(L,N), L=1,2), N=1,NTIM)
A. STEP = maximum allowable time step table, NTIM pairs of time step (sec) and problem time (sec).
22. SS
A. SS = thermal steady state switch.
23. QLIN
A. QLIN = total reactor power (W).
24. NOQ
A. NOQ = number of pairs in the power table (>1)
25. ((QDATA(L,N), L=1,2), N=1,NOQ)
A. QDATA = reactor power history table, NOQ pairs of relative reactor power and problem time (sec).

B.1.2 FLUIDS Module Input

B.1.2.1 Scalar Data

- 26. DTINC
 - A. DTINC = maximum allowable fractional timestep increase per step.
- 27. DTMIN
 - A. DTMIN = minimum timestep to cause problem termination.
- 28. CRFAC
 - A. CRFAC = Time step Courant factor.
- 29. ERROR1
 - A. ERROR1 = Convergence criterion on the change in relative pressure from iteration to iteration.
- 30. EPSA
 - A. EPSA = maximum fractional change in alpha over time step.
- 31. EPST
 - A. EPST = maximum fractional change in temperature over time step.
- 32. ITPT
 - A. ITPT = number of steps between title for short fluids print.
- 33. ITPTS
 - A. ITPTS = number of steps between short fluids print.
- 34. TINTPLT
 - A. TINTPLT = not currently used.

B.1.2.2 Array Data

- 35. ((ADKZ(J,I), J=1,KMAX),I=1,NRING)
 - A. ADKZ = additive axial friction factor applied at top of axial node J in Ring I for field 1.
- 36. ((ADKZ(J,I), J=1,KMAX),I=1,NRING)
 - A. ADKZ = additive axial friction factor for field 2.
- 37. ((ADKZ(J,I), J=1,KMAX),I=1,NRING)
 - A. ADKZ = additive axial friction factor for field 3.
- 38. ((ADKZ(J,I), J=1,KMAX),I=1,NRING)
 - A. ADKZ = additive axial friction factor for

- field 4.
39. ((ADKR(J,I), J=1,KMAX),I=1,NRING)
 A. ADKR = additive radial friction factor applied
 at right hand face of axial node J in
 Ring I for field 1.
 40. ((ADKR(J,I), J=1,KMAX),I=1,NRING)
 A. ADKR = additive radial friction factor for
 field 2.
 41. ((ADKR(J,I), J=1,KMAX),I=1,NRING)
 A. ADKR = additive radial friction factor for
 field 3.
 42. ((ADKR(J,I), J=1,KMAX),I=1,NRING)
 A. ADKR = additive radial friction factor for
 field 4.

B.1.2.3 Boundary Conditions

43. NINBC
 A. NINBC = number of locations for inflow boundary
 condition's (<6). If linked to TRAC set
 to 0.
44. INN
 A. INN = radial ring number of inflow boundary
 condition cell.
45. JIN
 A. JIN = axial node number of inflow boundary
 condition cell.
46. ARIN
 A. ARIN = flow area at the inflow boundary (m**2).
47. NPRIN
 A. NPRIN = number of entries in inflow pressure
 condition tables (<100).
48. NVIN
 A. NVIN = number of entries in inflow boundary
 condition velocity tables (<100).
49. NTIN
 A. NTIN = number of entries in inflow boundary
 condition temperature tables (<100).
50. NAIN
 A. NAIN = number of entries in inflow boundary
 condition alpha tables (<100).
51. ((PRNTAB(L,N), L=1,2), N=1,NPRIN)
 A. PRNTAB = inflow total pressure table, NPRIN pairs
 of time (sec) and pressure (Pa).

52. ((PH2TAB(L,N), L=1,2), N=1,NPRIN)
 A. PH2TAB = inflow hydrogen partial pressure table, NPRIN pairs of time (sec) and pressure (Pa).
53. ((VINTAB(L,N), L=1,2), N=1,NVIN)
 A. VINTAB = inflow boundary condition velocity table for a field.
54. ((TINTAB(L,N), L=1,2), N=1,NTIN)
 A. TINTAB = inflow boundary condition temperature table for a field.
55. ((AINTAB(L,N), L=1,2), N=1,NAIN)
 A. AINTAB = inflow boundary condition alpha table for a field.
56. NPBC
 A. NPBC = number of locations for outlet pressure boundary condition's (<6).
57. MPBC
 A. MPBC = Boundary condition location flag.
58. MOUT
 A. MOUT = Node for pressure outflow boundary.
59. AROUT
 A. AROUT = outflow area (m**2).
60. HDOUT
 A. HDOUT = outflow hydraulic diameter (m).
61. NPROUT
 A. NPROUT = number of entries in pressure outflow boundary condition table (<100).
62. ((PROTAB(L,N), L=1,2), N=1,NPROUT)
 A. PROTAB = outflow pressure boundary condition table.

B.1.2.4 Initial Conditions

63. PIN
 A. PIN = initial total pressure (spatially uniform) (Pa).
64. PINH2
 A. PINH2 = initial hydrogen partial pressure (spatially uniform) (Pa).
65. DH2IN
 A. DH2IN = initial hydrogen diffusion coefficient.
66. (ALIN(K), K=1,4)
 A. ALIN = initial volume fraction for field K.
67. (TIN(K), K=1,4)
 A. TIN = initial temperature for field K (K).

68. (VIN(K), K=1,4)
 A. VIN = initial axial velocity for field K (m/s).
69. (VRIN(K), K=1,4)
 A. VRIN = initial radial velocity for field K (m/s).
70. AL10
 A. AL10 = low alpha limit for all fields.
71. ICAVITY
 A. ICAVITY = cavity cell flag (=0 no cavity cell, =1 cavity cell).
72. JCAVITY
 A. JCAVITY = cavity cell number.
73. PCAVITY
 A. PCAVITY = pressure in cavity cell (Pa).
74. (TCAVITY(K),K=1,4)
 A. TCAVITY = temperature of field I in cavity cell (K).
75. (ACAVITY(K),K=1,4)
 A. ACAVITY = volume fraction of field I in cavity cell.

B.1.2.5 Geometry Data

76. (DZ(J), J=1,KMAX)
 A. DZ = length of each mesh cell (m).
77. (RA(I), I=1,NRING+1)
 A. RA = radial position of right edge of ring I (m).
78. IMBED
 A. IMBED = number of imbedded interfaces.
79. NRIMB
 A. NRIMB = radial node number for imbedded interface.
80. NZTIMB, NZBIMB
 A. NZTIMB = axial node number of cell at top of imbedded interface.
 B. NZBIMB = axial node number of cell at bottom of imbedded interface.
81. FAZIMB
 A. FAZIMB = flow area of imbedded interface (m**2).
82. ADKIMB
 A. ADKIMB = friction factor of imbedded interface.

B.1.3 CORE Module Input

B.1.3.1 Scalar Data and Power Shape

83. MODLSW, MXCMOD, ICPAHS, ICANDL, ICMECH, ICOXID, PHCNVG, CDCNVG, CMCNVG, OCCNVG, BETA1, FDLTM1, FDLTM2, FDLTM3
- A. MODLSW = Core module switch (0/1 = off/on) (cards 84, 85, and 88 must be input even if MODLSW=0).
 - B. MXCMOD = Maximum number of core structure models to be input in section A.3.2. (This is an estimate which should error on the high side.)
 - C. ICPHAS = Maximum number of iterations in phase change calculations (25).
 - D. ICANDL = Maximum number of iterations in candling calculations (100).
 - E. ICMECH = Maximum number of iterations in rod mechanics calculations. (Not implemented at this time.) (10)
 - F. ICOXID = Maximum number of iterations in oxidation calculations (Not implemented.) (10)
 - G. PHCNVG = Phase change convergence criterion. (1.0e-6)
 - H. CDCNVG = Candling convergence criterion. (1.0e-6)
 - I. CMCNVG = Mechanics convergence criterion. (1.0e-6)
 - J. OCCNVG = Oxidation convergence criterion. (1.0e-6)
 - K. BETA1 = Smoothing parameter. (Not implemented.) (0.0)
 - L. FDLTM1 = Remeshing parameter. (combining with neighbor).
 - M. FDLTM2 = Remeshing parameter. (dividing in two).
 - N. FDLTM3 = Remeshing parameter. (deleting from calculation).
84. ((CPSHP(J,I),J=1,KMAX),I=1,NRING)
- A. CPSHAP = Power shape for each fluids cell.

B.1.3.2 Model Definition Data

85. MDLNUM, MDTYPE, NUMAXR
- A. MDLNUM = Core model number.
 - B. MDTYPE = Type of core model.
 - C. NUMAXR = Number of axial submodels.

86. MDTPZ, IREGNS, ICHTYP, ICBND, ICPOW, MXCNOD, AXLEN, SMDLTZ
- A. MDTPZ = Submodel type (not currently used).
 - B. IREGNS = Number of radial regions to be input.
 - C. ICHTYP = Heat transfer option.
 - D. ICBND = Heat transfer boundary condition option (not currently used).
 - E. ICPOW = Power calculation option (not currently used).
 - F. MXCNOD = Maximum number of heat transfer nodes that could be used for all radial regions at an axial level.
 - G. AXLEN = Axial dimension of submodel section. (m)
 - H. SMDLTZ = Maximum axial node height in this submodel.
87. MDTYPR, ICNODS, ICMAT, NODESR, ICGAPR, RINNER, ROUTER, RPWFAC, FACTR1, FACTR2, FACTR3
- A. MDTYPR = Radial region type.
 - B. ICNODS = Remesh parameter that controls division of nodes.
 - C. ICMAT = Material property identification number.
 - D. NODESR = Number of heat transfer nodes in this region.
 - E. ICGAPR = Gap flag for gap on outer surface of this region.
 - F. RINNER = Inner radius of region. (m)
 - G. ROUTER = Outer radius of region. (m)
 - H. RPWFAC = Power factor for region. (≥ 0.0)
 - I. FACTR1 = Reserved for later use. (0.0)
 - J. FACTR2 = Reserved for later use. (0.0)
 - K. FACTR3 = Reserved for later use. (0.0)

B.1.3.3 Model Location Data

88. NUMMOD
- A. NUMMOD = Number of models in ring I (may be 0)
89. MDLNUM, CNMREP, PITCH, CPWFAC, BOTTOM, BURNUP, CTEMP, FGPRES.
- A. MDLNUM = Model number from card 85 input.
 - B. CNMREP = Number of actual structures to be represented by this model in this ring.
 - C. PITCH = Structure pitch. (m)
 - D. CPWFAC = Structure power factor. (≥ 0.0)
 - E. BOTTOM = Location of bottom of structure from problem bottom. (m)

- F. BURNUP = Fissionable material burnup. (Mwd/mT)
- G. CTEMP = Structure initial temperature. (K)
- H. FGPRES = Fill gas pressure. (Pa)

B.1.4 STRUCTURES Module Input

B.1.4.1 Scalar Data

90. MAXMOD, STCNVG, SFCNVG, ITCRST, ITMELT, IFAILS, NMATMX
- A. MAXMOD = number of unique structure models to be input.
 - B. STCNVG = freezing and melting convergence criterion.
 - C. SFCNVG = crust freezing and melting convergence criterion.
 - D. ITCRST = maximum iterations on crust.
 - E. ITMELT = maximum iterations on structure melting.
 - F. IFAILS = structure mechanics calculation switch (0=off, 1=on).
 - G. NMATMX = maximum number of materials for which mechanical properties are input on file MELMCH.

B.1.4.2 Model Definition and Boundary Conditions

91. ISTYPE, IHTTYP, MCHMAT, MCHMOD, IBNDOP, IPOWOP, NREGNS, MAXNOD, IDCEL, IDSUP, STTEMP
- A. ISTYPE = structure type identification number.
 - B. IHTTYP = heat transfer model switch.
 - C. MCHMAT = mechanics material property identification number.
 - D. MCHMOD = mechanical model section switch.
 - E. IBNDOP = heat transfer boundary condition flag.
 - F. IPOWOP = power option flag.
 - G. NREGNS = number of structure material regions to be input.
 - H. MAXNOD = maximum number of temperature nodes to be divided among all material regions.
 - I. IDCEL = unique identification number for structure.
 - J. IDSUP = unique identification number for the structure supporting structure IDCEL.
 - K. STTEMP = initial structure temperature (K).

92. STBND, STHTCF

- A. STBND = boundary sink temperature (K).
- B. STHTCF = boundary heat transfer coefficient (W/m**2/K).

B.1.4.3 Model Location and Geometry Data

93. ZBOT, [RIN, ROUT, ZTOP]

- A. ZBOT = location of the bottom of structure in the fluids mesh.
- BC RIN = inner radius of region over which the structure will be inserted in the fluids mesh.
- C. ROUT = outer radius of region over which the structure will be inserted in the fluids mesh.
- D. ZTOP = location of the top of the structure in the fluids mesh.

94. MATID, NNODES, XXXX

- A. MATID = region material thermal property identification number.
- B. NNODES = number of finite difference nodes in region.
- C. XXXX = the number of geometry input values varies with the type of structure.

a. For ISTYPE = 10-29, wall data.

RADIN, RADOUT, ZLENGTH

RADIN = inside radius of wall (m).

RADOUT = outside radius of wall (m).

ZLENGTH = length of wall (m).

b. For ISTYPE = 30-49, plate data.

The following input is for MCHMOD not = 2

RADIN, RADOUT, THICK, HOLED, HPTCH, COLMOD,
COLPTCH

RADIN = inside radius of plate (m).

RADOUT = outside radius of plate (m).

THICK = plate thickness (m).

HOLED = hole diameter for perforated plate (m).

HPTCH = hole pitch (m).

COLMOD = column outside diameter for plates supported on columns (m).

COLPTCH = column pitch (m).

The following input is for MCHMOD = 2

RADIN, RADOUT, THICK, HOLED, HPTCH, RMINOR, DUMMY

RADIN = inside radius of plate (m).

RADOUT = outside radius of plate (m).

THICK = plate thickness (m).

HOLED = hole diameter for perforated plate (m)

HPTCH = hole pitch (m).

RMINOR = minor vertical axis length for the
entire plate (m).

DUMMY = dummy variable (enter 0.0)

c. For ISTYPE = 50, column data

COLMIR, COLMOR, XLNGTH, COLMP

COLMIR = column inner radius (m).

COLMOR = column outer radius (m).

XLNGTH = length of column (m).

COLMP = column pitch (m).

d. For ISTYPE = 51, wall-plate joint model

RADIN, RADOUT, THICK

RADIN = inside wall radius (m).

RADOUT = outside wall radius (m).

THICK = plate thickness at joint (m).

e. For ISTYPE = 52, bolt model

BOLTOD, XLNGTH, THRDTP, CRCLOD, SPACE

TOLTOD = bolt shank diameter (m).

XLNGTH = bolt shank length (m).

THRDTP = thread type.

CRCLOD = diameter of circle on which bolts are
placed (m).

SPACE = bolt spacing along circle parameter (m).

f. For ISTYPE = 53, general structure (heat transfer
only).

ATOT, THICK

ATOT = total surface area (m^2).

THICK = thickness (m).

B.1.5 RADIATION Module Input

95. NGROUP

A. NGROUP = number of radiation groups (1).

96. ITRMAX

A. ITRMAX = max number of iterations (50).

- 97. RCONVA
 - A. RCONV = radiation convergence criterion (1.0E-4).
- 98. TBOUND
 - A. TBOUND = radiation boundary sink temperature (K).
- 99. (EMISS(K), N=1,6)
 - A. EMISS = emissivity of structure K. The structures are: pins, outer wall, inner wall, bottom plate, top plate, and debris.
- 100. RFAC, ARHOL, ARHOC, ARHOM
 - A. RFAC = steam absorption coefficient multiplier.
 - B. ARHOL = liquid absorption coefficient.
 - C. ARHOC = solid corium absorption coefficient.
 - D. ARHOM = liquid corium absorption coefficient.

B.1.6 DEBRIS Module Input

- 101. LDBRIS
 - A. LDBRIS = debris bed off/on (0/1) switch.
- 102. NZMAX, NBEDM, ICOND
 - A. NZMAX = maximum number of nodes in a bed (40-60).
 - B. NBEDM = maximum number of beds.
 - C. ICOND = effective conductivity model indicator.
 - a. ICOND = 1 means Imura-Takegoshi/Vortmeyer model
 - b. ICOND = 2 means use Willhite-Kunii-Smith/Luikov model.
- 103. PORMAX, DZMIN, DEFF, SO
 - A. PORMAX = maximum porosity in a bed (0.55).
 - B. DZMIN = minimum mesh size (0.05 m).
 - C. DEFF = effective particle diameter (0.01 m).
 - D. SO = initial specific power (w/kg). This should be set to the total initial power divided by the total mass of UO_2 .

B.1.7 Input for the Mechanics Calculation (at initialization only) on File MELMCH

- 1. NPTS
 - A. NPTS = Number of points in Young's modulus table (>1).
- 2. (TEMP1(N), N=1, NPTS)
 - A. TEMP1 = Temperature point in table (K).
- 3. (E(I), N=1, NPTS)
 - A. E = Young's modulus corresponding to temperature point (Pa).

4. NPTS
A. NPTS = Number of points in yield strength table (>1).
5. (TEMP1(N),N=1,NPTS)
A. TEMP1 = Temperature point in table (K).
6. (SY(N),N=1,NPTS)
A. SY = Yield strength corresponding to temperature point (Pa).
7. NPTS
A. NPTS = Number of points in ultimate tensile strength table (>1).
8. (TEMP1(N),N=1,NPTS)
A. TEMP1 = Temperature point in table (K)
9. (SU(N),N=1,NPTS)
A. SU = Ultimate tensile strength corresponding to temperature point (Pa).
10. NPTS
A. NPTS = Number of points in Poisson's ratio property table (>1).
11. (TEMP1(N),N=1,NPTS)
A. TEMP1 = Temperature point in table (K).
12. (NU(N),N=1,NPTS)
A. NU = Poisson's ratio corresponding to temperature point.
13. NPTS
A. NPTS = Number of points in thermal expansion coefficient table (>1).
14. (TEMP1(N),N=1,NPTS)
A. TEMP1 = Temperature point in table (K).
15. (ALF(N),N=1,NPTS)
A. ALF = Thermal expansion coefficient corresponding to temperature point (1/K).
16. IT,IS
A. IT = Number of temperatures in rupture table
B. IS = Number of failure stresses in rupture table.
17. (TEMP2(N),N=1,IT)
A. TEMP2 = Temperature point in rupture table (K).
18. (SIGMA(N),N=1,IS)
A. SIGMA = Failure stress point in rupture table (Pa).
19. ((RUPTIM(N,M),N=1,IT),M=1,IS)
A. RUPTIM = Rupture time corresponding to temperature and applied stress (sec).

C.0 GRAPHICS PROCESSING

Graphics output from MOD1 is produced on file "CMOUT." This file contains mostly fluids variables from the FLUIDS module, plus surface temperatures for rod types 1 and 2 from the CORE module. There are also several global variables, such as entrance and exit mass. The file is produced by routine GRF2D in a format suitable for postprocessing by the TRAP program. TRAP is the graphics postprocessor for the TRAC reactor systems code; normally, TRAC produces a graphics file (TRCGRF) that must be processed by another program (EXCON), which generates separate files for each requested TRAC component; these "component" files are then processed by TRAP. CMOUT, however, is in what is called "unpacked component format," and can be read directly by TRAP.

The following sections describe the CMOUT file format and the procedure for using TRAP (on CTSS), and give a few examples of TRAP output.

C.1 CMOUT Graphics File Structure

The CMOUT graphics file has the same logical structure as a TRAP component file; specifically, it is a component file of type "UVESSEL," which designates a vessel component in unpacked format to TRAP. The file structure is divided into three sections (see Table C.1), which are the P header block, the Q header block, and the time edit block. The two header blocks are output once at the start of every MELPROG steady-state or restart run. The time edit block is output at each graphics output time (set by input parameter TPLT) and contains the values of the output variables at the current time.

The P block header contains data (Table C.2) that is fixed for the duration of the computer run, and is independent of the type of component (this is for compatibility with TRAP; MOD1 presently produces only one type of component graphics file). The P block contains the start time, end time, and problem title, all from the MOD1 input file.

The Q block header (Table C.3) contains parameters that vary with this type of component; here, the parameters are those for a vessel component. Referring to Table C.3, there are some parameters that depend both on the individual problem geometry and on the specific variables contained in the graphics file, and some that depend only on one or the other. NDEPV, for instance,

Table C.1		
CMOUT File Structure		
<u>Section</u>	<u>Words</u>	<u>Contents</u>
BOCBOC	1	Beginning-of-Component Symbol
P Block Header	19	Fixed Header (start, end times, title)
Q Block Header		Variable Header for vessel component
EOHEOH	1	End-of-Header Symbol
First Time Edit		Data for First Time Edit
EOTEOT	1	End-of-Time_Edit Symbol
.		
.		
.		
Last Time Edit		Data for Last Time Edit
EOTEOT		
EOCEOC	1	End-of-Component Symbol

Table C.2		
Structure of P Block Header		
<u>Section</u>	<u>Words</u>	<u>Contents</u>
File Name	1	Name of Component; for CMOUT, this is "COMP01".
Start Time	1	Starting Time of Problem Run
End Time	1	End Time of Problem Run (from input, not necessarily how long the run actually is).
Title	16	Problem title in Hollerith, each word is 8 characters (bytes) long.

is the number of dependent variables in each time edit on the file. There are NDEPV names, which are the names used to refer to the output variables in the TRAP command file, and NDEPV LABELS, which are the labels for the dependent variables which appear on the graphs produced by TRAP. The parameters that depend on the number of dependent variables are the ones that a programmer adding output variables will want to change. The other parameters will change automatically and require no programmer attention.

Contained within the Q block are two component-dependent blocks, LCDATA and Geometry shown in Table C.3. These contain data for a vessel component in CMOUT. Table C.4 shows the LCDATA contents, and Table C.5 shows the arrangement of the data in the Geometry block.

The structure of the Time Edit block is shown in Table C.6. Some points to note are:

1. The global variables (those not having any spatial dependence) come first, followed by the spatially dependent variables; the time variable, although shown first, can actually occur anywhere in the global area.
2. The time is repeated at the start of the spatial region, with a NAME "timet;" although the length NWRDS of this variable is 1, the offset IPTR is NCELLS. This is a TRAP peculiarity, but is noted here for anyone wishing to read or modify the CMOUT file.
3. The NAME of the first spatially dependent variable must be "alpha;" this is again a TRAP peculiarity.

C.2 TRAP Usage on CTSS

TRAP is a DISSPLA-based program written in FORTRAN and can be linked to a variety of output devices, or can produce a device-independent output file called a metafile, which can then be postprocessed and sent to the desired output device. TRAP expects the graphics input file (MOD1 file CMOUT) to be in the local file area and to have a name like "compnn," where nn is a number such as 01, 20, etc. The input file name, as well as graphics commands to TRAP, can be either input interactively from a terminal or contained in an ASCII command file. The graphics

Table C.3		
Structure of Q Block Header		
<u>Section</u>	<u>Words</u>	<u>Contents</u>
NDEPV	1	Number of dependent variables on file, plus 1 for time.
LTIME	1	Length of dependent variable data in a time edit block, not including the EOTEOT symbol. In MOD1, $LTIME = KMAX * NRING * NVAR + NGLOB + 2$, where $NGLOB$ = number of global variables, $NVAR$ = number of spatially dependent variables, and the 2 is because time is repeated twice.
LDLEN	1	Length of the vessel level data. In MOD1, $LDLEN = NRING * NVAR$.
INUM	1	User assigned component ID number (1).
TYPE	1	Component type (vessel, pipe, etc.); in MOD1, this is "UVESSEL".
ID	1	Computer code assigned ID number (1).
NCELLS	1	Number of cells in component; for MOD1, equal to $NRING$.
.		
.		
.		
LCDATA Block	varies	Additional data that varies with the type of component; in MOD1, this block is that for a VESSEL (see Table C.4).
Geometry Block	varies	Geometry data, varies with type of component (MOD1 uses VESSEL).

Table C.3 (cont.)		
Structure of Q Block Header		
<u>Section</u>	<u>Words</u>	<u>Contents</u>
NAME ₁ -NAME _{NDEPV}	NDEPV	Dependent variable names, used to refer to variables in TRAP. These are Holleriths, 8 bytes each.
IPTR ₁ -IPTR _{NDEPV}	NDEPV	Dependent variable pointers, point to relative location of dependent variables from start of time edit block
NWRDS ₁ -NWRDS _{NDEPV}	NDEPV	Length of dependent variable/level (e.g., 1 for time, NCELLS for pressure).
LABEL ₁ -LABEL _{NDEPV}	3*NDEPV	Labels for dependent variables to be used on graphs, each three words long (Holleriths, 24 bytes each).

Table C.4		
Structure of LCDATA Block for CMOUT		
<u>Section</u>	<u>Words</u>	<u>Contents</u>
NWORD	1	Number of words to follow in LCDATA Block (7).
NCLX	1	Number of cells/level (NRING).
NASX	1	Number of levels (KMAX).
NRST	1	Number of radial cells (NRING).
NTSX	1	Number of azimuthal cells (1).

Table C.4 (cont.)		
Structure of LCDATA Block for CMOUT		
<u>Section</u>	<u>Words</u>	<u>Contents</u>
NCSR	1	Number of source connections (pipes) into each cell (set to 1, not used).
NODES	1	Number of heat transfer nodes (set to 1, not used).
NZMAX	1	Maximum number of fine mesh axial cells (set to 1, not used).

Table C.5	
Structure of Geometry Block in CMOUT	
<u>Section</u>	<u>Contents</u>
DZ (1-NASX)	Axial lengths (x(1deltaz+i), i=1,kmax).
DR (1-NRSX)	Radial widths (x(1dr+i), i=1,nring).
DTH (1-NTSX)	Angular widths in radians (2π).
VOL ₁ (1-NCLX)	Cell volumes, first level.
FA ₁ (1-3*NCLX)	Flow areas, first level. These are arranged as axial flow area, radial flow area, and azimuthal flow area (set to 0 here).
.	
.	
.	
VOL _{NASX} (1-NCLX)	Cell volumes, last level.
FA _{NASX} (1-3*NCLX)	Flow areas, last level. These are arranged as axial flow area, radial flow area, and azimuthal flow area (set to 0 here).

Table C.5 (cont.)	
Structure of Geometry Block in CMOUT	
<u>Section</u>	<u>Contents</u>
ISRL (1-NCSR)	Source level numbers (not used, set to 1).
ISRC (1-NCSR)	Source cell numbers (not used, set to 1).
ISRF (1-NCSR)	Source face numbers (not used, set to 1).
ICO (1-NCSR)	Adjacent component ID (not used, set to 1).
NSRL (1-NASX)	Number of sources/level (not used, set to 1).

Table C.6		
Structure of Time Edit Block		
<u>Section</u>	<u>Words</u>	<u>Contents</u>
TIME	1	Time
GLOBAL ₁	1	First global variable
.		
.		
GLOBAL _{NGLOB}	1	Last global variable
TIMET	1	Time (again, see text).
SPACE _{1,1}	NRING	First spatial variable, first level, for all cells (rings) on that level.
SPACE _{2,1}	NRING	Second spatial variable, first level, for all cells (rings) on that level.
.		
.		
SPACE _{NVAR,1}	NRING	Last spatial variable, first level, for all cells (rings) on that level.
.		
.		

Table C.6 (cont.)		
Structure of Time Edit Block		
<u>Section</u>	<u>Words</u>	<u>Contents</u>
SPACE _{1,NASX}	NRING	First spatial variable, last level, for all cells (rings) on that level.
SPACE _{2,NASX}	NRING	Second spatial variable, last level, for all cells (rings) on that level.
.		
.		
SPACE _{NVAR,NASX}	NRING	Last spatial variable, last level, for all cells (rings) on that level.

output file produced by TRAP is either file tape77 (if TRAP was linked to a specific output device), or tape55 (if TRAP produces a metafile).

A few points: TRAP must be linked while running CCL under CTSS; for instance, the command

```
/ diss met
```

will produce an executable file called xdissabs from a local \$bld file at SNLA; xdissabs is a metafile version of TRAP, which can be renamed to something more mnemonic, such as "trapmet." Giving the command

```
trapmet
```

will then start TRAP in interactive mode, that is, the control commands are typed in from the keyboard. TRAP will produce a file named "langfl" containing a text record of the interactive session; langfl may be edited into a command file and saved to repeat the same run later. Since this is a metafile version of TRAP, the output file is tape55, which may be postprocessed (by posti) to give a tape77 file suitable for output to a specific device, such as a Tektronics terminal or laser printer.

Another point to note is that to produce fancy labels on

graphs suitable for publication (using the "förmal" command), the desired labels must be on a file named "lblist" in the local area. A label file compatible with the output variables on the MOD1 graphics file is available from SNLA. The second line of each variable record in the lblist file is usable to produce graphs in alternate units from the standard SI units, for instance, English engineering units. The lblist file at SNLA does not contain any conversion factors to do this, and the second line of each record is the same as the first.

D.0 HEAT TRANSFER AND FLOW CORRELATIONS

The field equations in the FLUIDS module, Eqs.(2.1) through (2.8), require certain auxiliary or constitutive equations to achieve closure. For example, equations for wall shear, interfacial drag, interfacial heat transfer and wall heat transfer must be specified. Detailed descriptions of these correlations are provided in this appendix for the vapor and liquid fields. The current version of the FLUIDS module neglects heat transfer and drag between walls and the corium fields. However, such correlations for these latter fields can be included if experiments indicate they are important.

The correlations discussed in this appendix are borrowed from the TRAC code [D.1]. Correlations for wall shear and wall heat transfer are given in Sections D.1 and D.3, respectively. Section D.2 discusses heat transfer and drag forces between the vapor and liquid fields.

D.1 Wall Shear

The total pressure gradient calculated in the momentum equations is expressed as the sum of the frictional dissipation, acceleration head, and potential head terms. Subroutine WFRIC1 calculates coefficients for the frictional dissipation terms and for losses associated with abrupt area changes. Under single-phase flow conditions, pressure drops associated with frictional losses are correlated as functions of fluid velocity, fluid density, fluid viscosity, channel hydraulic diameter, and surface roughness of the channel wall. When a two-phase mixture is flowing in a channel, a correction to the single-phase frictional loss is necessary to account for added dissipation between phases and interactions with the channel walls. This correction factor is the two-phase flow multiplier.

The wall shear coefficients for the vapor and water fields, c_{w1} and c_{w2} , are defined as

$$c_{w1} = \alpha_1 \rho_1 \frac{c_{f1}}{D_h} \quad (D.1)$$

and

$$c_{w2} = \alpha_2 \rho_2 \frac{c_{f2}}{D_h} \quad ; \quad (D.2)$$

where c_{f1} and c_{f2} are the vapor and liquid friction factors (these are given below); α is a volume fraction; ρ is density; D_h is a hydraulic diameter; and the subscripts 1 and 2 refer to vapor and liquid, respectively. The WFRIC1 routine allows several options to calculate wall friction, including constant value (user input), a homogeneous model, and calculation of an appropriate form-loss coefficient in addition to the selected two-phase flow friction factor if there are abrupt area changes. MELPROG/MOD1 currently uses only the homogeneous model.

The homogeneous friction-factor model alters the single-phase value by using a two-phase viscosity ($\bar{\mu}$) defined in terms of the flow quality, x [D.2]:

$$\frac{1}{\bar{\mu}} = \frac{x}{\mu_g} + \frac{(1-x)}{\mu_l} \quad (D.3)$$

The homogeneous friction factor is then given by [D.3]:

$$f = 0.046 (\text{Re})^{-0.2} \quad \text{if } \text{Re} \geq 5000 ; \quad (D.4a)$$

$$f = 0.032 - 5.25 \times 10^{-6} (\text{Re} - 500) \quad \text{if } 500 < \text{Re} < 5000 ; \quad (D.4b)$$

$$f = 0.032 \quad \text{if } \text{Re} \leq 500 ; \quad (D.4c)$$

where $\text{Re} = GD_h/\bar{\mu}$ and $G = \rho_m V_m$. These equations represent constant friction factors for $\text{Re} \leq 500$ and $\text{Re} \geq 5000$ with a linear interpolation between $500 < \text{Re} < 5000$. The coefficient of friction for the liquid is

$$c_{f2} = 2.0 f , \quad (D.5)$$

and, if the void fraction is greater than 0.90, the vapor coefficient of friction is

$$c_{f1} = (10.0\alpha_1 - 9.0)^2 (21.0 - 20.0\alpha_1) c_{f2} \quad (D.6)$$

The purpose of this function is to ensure a smooth transition from zero-vapor wall friction at $\alpha = 0.9$ to the single-phase vapor value at $\alpha = 1.0$.

Separate wall friction coefficients are calculated for the axial and radial directions. These coefficients are calculated using edge-cell properties; the cell-centered properties (such as density and volume fraction) are averaged to get edge-cell values, and the axial or radial velocities are used for the axial

or radial cell faces, respectively. Hydraulic diameters and flow areas on the cell faces account for the presence of walls or plates with orifices (diameters and flow areas are calculated in routine STRSET in the STRUCTURES module). The user can specify additional constant friction factors and flow areas to account for geometric details whose scale is smaller than the mesh-cell size.

D.2 Interfacial Heat Transfer and Shear

The interfacial constitutive equations are basically extended versions of those in TRAC-PF1/MOD1. The interfacial heat transfer during boiling and the interfacial shear are calculated in conjunction with a simple flow-regime map [D.4]. This flow-regime map, although originally developed for vertical pipe flow, is the simplest prescription that provides a rational means for defining the constitutive equations. Figure 2.1 in Chapter 2 illustrates how the flow map is implemented in the code.

D.2.1 Void Fractions Less Than 0.3

A bubbly flow is assumed in this regime and the interfacial surface area is calculated in conjunction with a critical bubble Weber number, We_b . A value of $We_b = 7.5$ is used in MELPROG/MOD1. This choice was based on comparisons between TRAC predictions and experimental results for low subcooling (i.e., shear-dominated) downcomer tests performed by Creare, Inc. [D.5]. The expression relating interfacial surface area to We_b is

$$\frac{\rho_2 v_r^2 D_b}{\sigma} = We_b \quad (D.7)$$

or

$$D_b = \frac{We_b \sigma}{\rho_2 v_r^2} \quad , \quad (D.8)$$

where D_b is the bubble diameter, σ is the surface tension and v_r is the relative velocity between fields 1 and 2. The bubble diameter must lie between the mesh-cell hydraulic diameter and 10^{-4} m. For this diameter, with the assumption of a uniform bubble distribution within the mesh-cell volume, the number of bubbles is

$$\text{CNB} = \frac{6\alpha_1 V}{\pi D_b^3}, \quad (\text{D.9})$$

and the interfacial area is

$$A_{12} = 6\alpha_1 V \rho_2 \frac{v_r^2}{We_b \sigma}, \quad (\text{D.10})$$

where V is the volume of the cell. If the relative velocity is very small, this area can become small enough to allow significant nonequilibrium. Another surface area,

$$A = 4.83\alpha_1^{2/3} (N_B)^{1/3} V, \quad (\text{D.11})$$

based on a minimum number density ($N_B = 10^{16}$ bubbles/m³), is computed, and the actual surface area used is the larger of the two.

The total heat transfer from vapor to liquid is the sum of two contributions: vapor to an interface at the saturation temperature, T_s , and this interface to the liquid. Heat transfer from the interface to the liquid is given by $h_{i2}(T_s - T_2)$, where the liquid-side interfacial heat transfer coefficient (HTC), h_{i2} , depends primarily on field 2 properties; and heat transfer from the vapor to the interface is $h_{i1}(T_1 - T_s)$, where h_{i1} is the vapor side HTC.

For boiling, h_{i2} is the larger of an approximate formulation of the Plesset-Zwick bubble growth model [D.6]:

$$\text{Nu} = \frac{12}{\pi} (T_2 - T_s) \rho_2 \frac{1}{\rho_1 L} \frac{\partial u_2}{\partial T_2}, \quad (\text{D.12})$$

where L is the latent heat of vaporization and u is the internal energy; and a sphere convection coefficient [D.7]:

$$\text{Nu} = 2.0 + 0.74 \text{Re}_b^{0.5}, \quad (\text{D.13})$$

where

$$\text{Re}_b = \rho_2 v_r \frac{D_b}{\mu_2}.$$

For subcooled liquid in a dispersed bubbly regime, the liquid-

side interfacial heat transfer coefficient is:

$$h_{i2} = 0.02 \rho_2 c_{p2} v_r \quad (D.14)$$

Equation (D.14) is for turbulent convection and is derived from the expression $Nu = 0.02Re$ by assuming that the Prandtl number is one and that $c_p = c_v$.

In the bubbly regime, the vapor to interface HTC is one of two constants. If the vapor is superheated, h_{i1} is 1,000; if it is subcooled, h_{i1} is 10,000.

The interfacial shear coefficient is provided by a standard formula for a sphere [D.8]:

$$c_{12} = \frac{3c_b \alpha_1 \rho_2}{4D_b} \quad ; \quad (D.15)$$

where

$$c_b = 240 \quad \text{if } Re_b < 0.1 \quad ;$$

$$c_b = \frac{24}{Re_b} \quad \text{if } 0.1 \leq Re_b \leq 2 \quad ;$$

or

$$c_b = \frac{18.7}{Re_b^{0.68}} \quad \text{if } Re_b > 2 \quad .$$

D.2.2 Void Fractions Between 0.3 and 0.5

If the vapor fraction is between 0.3 and 0.5, the flow enters the slug regime. At the maximum α of 0.5, 40% of the vapor is assumed to exist in the form of trailing bubbles with the remainder contained in the slug. These bubbles contribute the majority of the interfacial heat transfer because their surface area is greater than that for the slug. Therefore, the correlations given in the preceding section for entrained bubbles are used. In the slug regime the effective hydraulic diameter is determined by a linear weighting in α between the Weber number criterion and the channel hydraulic diameter such that the effective diameter is the hydraulic diameter at an α of 0.5 and the bubble diameter determined from the Weber number at an α of 0.3.

D.2.3 Void Fractions Between 0.50 and 0.75

For the regime between the droplet and the bubbly slug flow, a cubic spline interpolation in the vapor fraction is made between the conditions that would exist if the vapor fraction were 0.75 in the annular or annular-mist topology (see next section) and the conditions that would exist if the void fraction were 0.5 in the bubbly slug regime. This interpolation assures that the correlation for the interfacial shear, interfacial heat transfer, and surface area is a continuous function of the vapor fraction, the relative velocity, the mass flux, and the various fluid thermodynamic and transport properties.

D.2.4 Void Fractions Between 0.75 and 1.00

In the α range of 0.75 to 1.0, an annular or annular-mist regime is assumed. The liquid fraction, E , that exists as mist is obtained from the following correlation which combines a simple s-shaped entrainment correlation based on the critical Weber number and a correlation by Kataoka and Ishii [D.9]:

$$E = \max \left\{ 1 - \exp \left[\frac{(v_E - v_1)}{v_E} \right] 0.5, 7.75 \times 10^{-7} We_e (Re_2 We_e)^{1/4} \right\} \quad (D.16)$$

where

$$v_E = 2.33 \left[\frac{(\rho_2 - \rho_1) \sigma We_d}{\rho_1^2} \right]^{1/4},$$

$$We_e = \frac{\rho_1^{2/3} (\alpha_1 v_1)^2 D_h (\rho_2 - \rho_1)^{1/3}}{\sigma},$$

and Re_2 is the liquid Reynolds number ($Re_2 = \rho_2 v_2 D_h / \mu$).

The remainder of the liquid is in a film or sheet. The interfacial shear and heat transfer are volume averages of the film and droplet relations in the annular-mist regime. The wetted surface area of the mesh cell is determined from the rod or structure heat-transfer area in the cell. The total interfacial surface area is determined by the sum of the areas contained in the wetted film and the droplets. A critical Weber number of 4 for the drops is used with a calculation procedure similar to that for bubbly flow. This value of the Weber number is appropriate for accelerating drops. For those cases where

sensitivity to We_d was tested, the results were not influenced strongly by We_d in the range of $2 \leq We_d \leq 12$.

For superheated drops, the liquid-side HTC is simply

$$h_{i2} = \frac{ak_2}{D_d} \quad , \quad (D.17)$$

where the Nusselt number, a , is a constant that has been adjusted to drive the drops to equilibrium under a variety of flow conditions and D_d is the droplet diameter. In MELPROG/MOD1, $a = 15,000$, which implies a droplet thermal boundary layer thickness that is approximately one ten-thousandth of the drop diameter. For subcooled drops, h_{i2} is obtained by assuming circulation in the drop, giving

$$h_{i2} = 0.02 \rho_2 c_{p2} v_{\text{circ}} \quad , \quad (D.18)$$

where

$$v_{\text{circ}} = 0.5 \left[\frac{\mu_1}{\mu_1 + \mu_2} \right] v_r \quad . \quad (D.19)$$

At some point, this expression for v_{circ} produces centripetal source terms large enough compared to the surface tension to produce distortion and eventually to break up the drop. Hence, v_{circ} cannot exceed

$$v_{\text{circ}} = 1.4 \sqrt{\sigma / (D_d \rho_2)} \quad .$$

The heat transfer between the vapor and the droplet interface is obtained from

$$Nu = 2.0 + 0.74 Re_1^{1/2} Pr_1^{1/3} \quad . \quad (D.20)$$

For any wall film that may exist along with the drops, the liquid-side interfacial HTC is

$$h_{i2} = 0.02 \rho_2 c_{p2} v_2 \quad ; \quad (D.21)$$

and for the vapor side,

$$h_{i1} = 0.02 \rho_1 c_{p1} v_1 \quad . \quad (D.22)$$

The interfacial-drag coefficient for the annular-drop regime combines a droplet drag that is analogous to Eq.(D.15) with

Wallis' correlation for annular flow [D.10], giving:

$$c_i = \frac{0.01\rho_1}{D_h} \left(1.0 + 75.0 (1 - \alpha_1)(1 - E) \right) , \quad (D.23)$$

where E is given by Eq.(D.16). To avoid a singularity in the liquid acceleration, a void fraction, α_{cut} , is calculated that corresponds to the minimum allowed film thickness. For void fractions above this value, the above equation is multiplied by $(1 - \alpha_1)/(1 - \alpha_{cut})$. To obtain the total interfacial-drag coefficient, droplet drag is weighted by the liquid fraction that is entrained, and Wallis' annular flow is weighted by the fraction remaining as film.

D.3 Wall-to-Fluid Heat Transfer

The wall-to-fluid heat transfer coefficients are obtained from a generalized boiling curve constructed within subroutine HTCORM. Fig. D.1 shows a portion of the boiling curve, which is not dependent on the flow regime. The single-phase vapor and condensation regimes are not shown in this figure.

The individual correlations used for each heat-transfer regime and the method of partitioning the energy between the two phases are discussed in this section. The boiling curve should be continuous between regimes for best numerical convergence in the pressure iteration; thus, the curve has been smoothed using various methods.

D.3.1 Wall-to-Fluid HTC Selection Logic

The heat transfer coefficient selection logic is outlined in this section. Heat transfer coefficients for the regimes of interest are given in Section D.3.2. The following logical sequence is used (if the criteria in one step are not satisfied, then the next step is examined):

- Step 1. Calculate both the flow and the equilibrium qualities, as well as the slip (vapor velocity/liquid velocity). Correlations are not currently available for countercurrent flow (slip<0.0) or for situations in which the vapor velocity is less than the liquid velocity. Consequently, whenever the slip is less than one, it is set equal to 1.0.

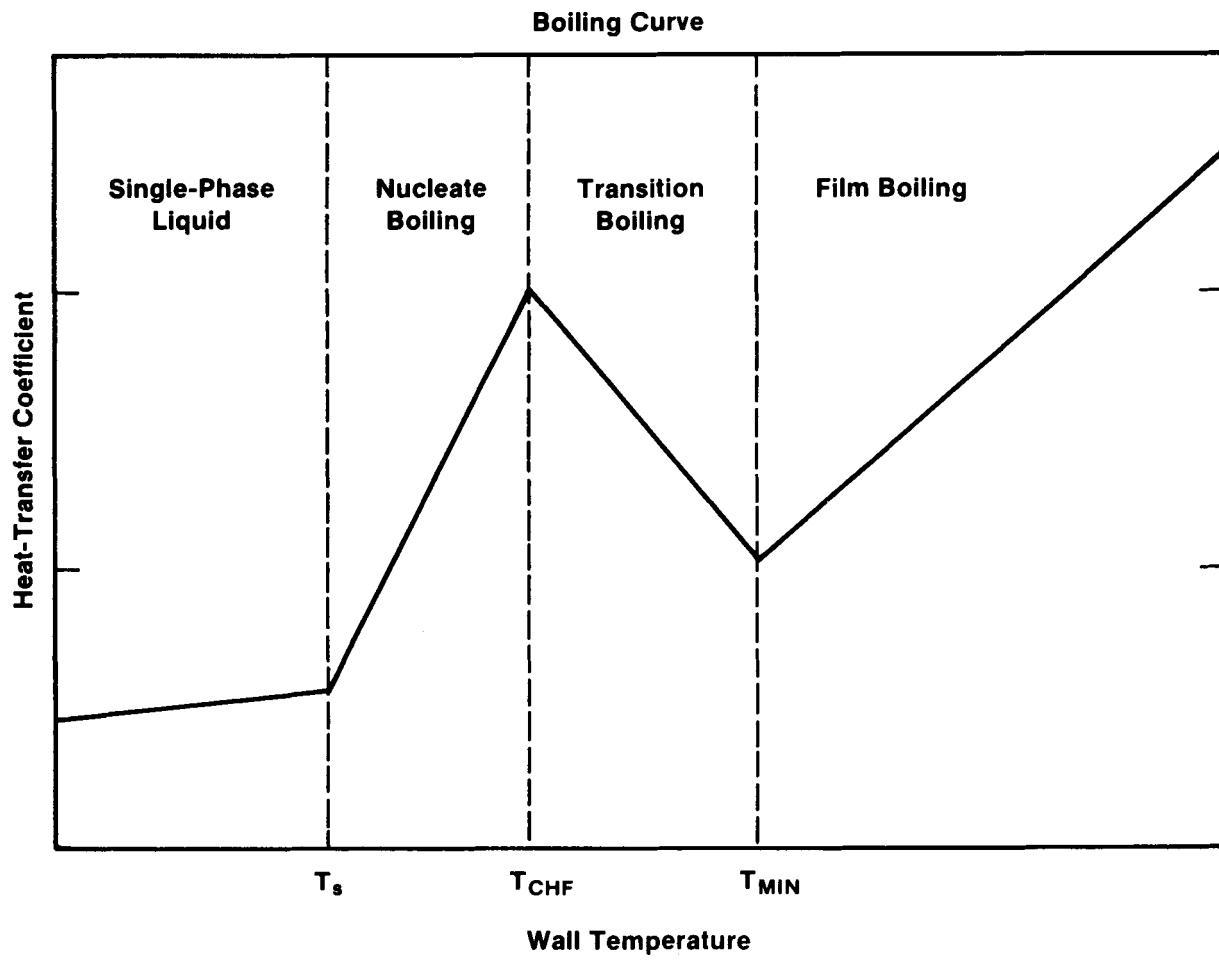


Fig. D.1 Boiling curve.

- Step 2. If $T_w < T_s$, $\alpha > 0.05$ and $T_w < T_1$; $T_w < T_s$, $\alpha > 0.05$, $T_w \geq T_1$, and $T_w < T_2$; or the liquid-vapor interface is present in the cell, the heat transfer is in the condensation regime. The Nusselt [D.2] and Chen [D.11] correlations, discussed in Section D.3.2.2, are used in this regime to calculate the vapor and liquid HTC's, respectively. If the equilibrium quality, x , is less than or equal to 0.71, $h_1 = 0$. For $x > 0.71$, h_1 is found by linear interpolation between zero and the value obtained from the Nusselt analysis [D.2]. The suppression factor, S , for the Chen correlation [D.11] is set to zero. If $x > 0.71$, the limit of Chen's data base, then the Chen correlation is evaluated at $x = 0.71$, which gives h_2 . Linear interpolation is used (between h_2 calculated corresponding to $x = 0.71$ and the single-phase vapor h_2 of zero) for an equilibrium quality exceeding 0.71.
- Step 3. If $0.96 < \alpha < 1.0$, a single-phase vapor HTC is calculated. When $\alpha > 0.98$, linear interpolation is used between the HTC's from the appropriate heat-transfer regime and the single-phase vapor values. The h_2 is interpolated between $0.98 \leq \alpha \leq 0.999$; the h_1 is interpolated between $0.98 \leq \alpha \leq 1.0$.
- Step 4. The forced-convection and nucleate-boiling HTC's are calculated by using the Chen correlation. At this point the heat-transfer regime has not been determined.
- Step 5. If $T_w < T_2$ or $T_w < T_s$, the heat-transfer regime is single-phase liquid.
- Step 6. The critical heat flux (CHF) and the corresponding temperature, T_{CHF} , are calculated next.
- Step 7. If $T_w < T_{CHF}$, nucleate boiling exists.
- Step 8. The minimum stable film-boiling temperature, T_{min} , is calculated by using the homogeneous-nucleation model.
- Step 9. If $T_w < T_{min}$, transition boiling exists. It previously has been determined that T_w is greater than the temperature at CHF conditions (step 8).
- Step 10. If $T_w > T_{min}$, film boiling is occurring.

Step 11. The HTC's are restricted to zero or positive values. (For $\alpha > 0.999$, extrapolation occurs and a negative h_2 could result. See step 3.) For $\alpha < 0.15$, the vapor heat flux is interpolated linearly from its value at $\alpha = 0.15$ to a value of 0.0 at $\alpha = 0.01$. The liquid heat flux is adjusted so that the total heat flux remains unchanged. Then, h_2 and h_1 are recalculated from the adjusted heat fluxes and the known temperature differences to prevent an extremely small mass of vapor from becoming superheated to an unrealistic value.

Step 12. The HTC's are averaged between the previous and the current time steps.

D.3.2 HTC Correlations

In this section we discuss the HTC correlations used in the construction of the boiling curve. Because MELPROG/MOD1 is a nonequilibrium code, HTC's are needed for the liquid and the vapor phases; the partitioning of energy between phases also is discussed. Table D.1 lists the heat-transfer regimes that are used.

TABLE D.1	
TRAC-PF1/MOD1 Heat-Transfer Regimes	
<u>IDREG</u> <u>(IHTF)</u>	<u>Wall-to-Fluid Heat-Transfer Regime</u>
1	Forced convection to single-phase liquid
2	Nucleate boiling
3	Transition boiling
4	Film boiling
6	Convection to single-phase vapor
11	Condensation
12	Liquid natural convection

D.3.2.1 Single-Phase Liquid (Heat-Transfer Regimes 1 and 12)

Either forced convection (regime 1) or natural convection (regime 12) can occur when single-phase liquid is present. Forced-convection correlations are used when the ratio of the Grashof number to the Reynolds number squared is less than or

equal to 1.0. Laminar or turbulent forced-convection correlations are available. These equations are contained in subroutine CHEN because they constitute one part of the Chen correlation [D.11]. The laminar equation is [D.3]:

$$h_2 = 4.0 \frac{k_2}{D_h} \quad (D.24)$$

The turbulent HTC is found from the Dittus-Boelter equation [D.12],

$$h_2 = 0.023 \frac{k_2}{D_h} Re_2^{0.8} Pr_2^{0.4} \quad (D.25)$$

where the liquid Reynolds number is

$$Re_2 = \frac{\rho_2 v_2 D_h}{\mu_2} \quad (D.26)$$

and the liquid Prandtl number is

$$Pr_2 = \left[\frac{\mu_{c,p}}{k} \right]_2 \quad (D.27)$$

All the properties are evaluated at T_2 . The maximum of Eqs.(D.24) and (D.25) is set equal to the single-phase liquid forced-convection HTC, HFORC.

In heat-transfer regime 12, natural convection to single-phase liquid, the maximum of the laminar and turbulent correlations for vertical flat plates and cylinders is used [D.13]. For laminar flow, the equation is

$$Nu = 0.59 (GrPr_2)^{0.25} \quad (D.28)$$

and for turbulent flow,

$$Nu = 0.10 (GrPr_2)^{1/3} \quad (D.29)$$

where the Grashof number is

$$Gr = \frac{g\beta |T_w - T_2| \rho_f^2 D_h^3}{\mu_2^2} \quad (D.30)$$

the Prandtl number is again given Eq.(D.27) and where the Nusselt number is given by:

$$Nu = \frac{h_2 D_h}{k_2} \quad . \quad (D.31)$$

The maximum of the laminar and turbulent Nusselt numbers is used. All the properties are evaluated at T_2 except the density, ρ_2 , and the volume coefficient of expansion, β ; these are evaluated by using a Taylor-series expansion about T_2 :

$$\rho_f = \rho_2 + \frac{\partial \rho_2}{\partial T} (T_f - T_2) \quad , \quad (D.32)$$

$$\beta = - \frac{\partial \rho_2}{\partial T} \frac{1}{\rho_f} \quad , \quad (D.33)$$

and

$$T_f = \frac{1}{2} (T_w + T_2) \quad . \quad (D.34)$$

Two points should be noted about heat-transfer regimes 1 and 12. First, because only single-phase liquid is assumed to be present, the vapor HTC, h_1 , is set equal to zero. Second, in Eq.(D.28) the hydraulic diameter, D_h , is used for the characteristic length even though the axial distance would be more appropriate, because the axial length would approach zero near the bottom of a channel. The characteristic-length term drops out of Eq.(D.29) and appears only to the 0.25 power in Eq.(D.28). Thus, the choice of the characteristic length has only a small effect on the HTC.

D.3.2.2 Nucleate-Boiling HTCs (Heat-Transfer Regime 2)

The Chen correlation [D.11] is used in heat-transfer regime 2, nucleate boiling. In addition, the nucleate-boiling HTC correlation affects the transition-boiling regime through the interpolation between the CHF and the minimum stable film-boiling points. The Chen correlation has two parts, a forced-convection term multiplied by the Reynolds factor, F , and a nucleate-boiling term that contains the factor S (both F and S are given in graphical form by Chen), and is given by:

$$h_2 = F h_{forc} + \min \left(1, \frac{T_w - T_s}{T_w - T_2} \right) S h_{nucb} \quad , \quad (D.35)$$

where h_{forc} is calculated in the manner discussed in the preceding section, h_{nucb} is given by:

$$h_{nucb} = 0.00122 \frac{k_2^{0.79} c_{p2}^{0.45} \rho_2^{0.49}}{\sigma^{0.5} \mu_2^{0.29} L^{0.24} \rho_1^{0.24}} (T_w - T_s)^{0.24} (p_w - p)^{0.75}, \quad (D.36)$$

L is the latent heat and p_w is the saturation pressure corresponding to the wall temperature. The Reynolds number factor, F , can be expressed as [D.14]:

$$F = 1.0 \quad \text{if } \chi_{TT} \leq 0.10; \quad (D.37)$$

and

$$F = 2.35(\chi_{TT}^{-1} + 0.213)^{0.736} \quad \text{if } \chi_{TT}^{-1} > 0.10; \quad (D.38)$$

where χ_{TT}^{-1} , the Lockhart-Martinelli factor, is

$$\chi_{TT}^{-1} = \left(\frac{x}{1-x} \right)^{0.9} \left(\frac{\rho_2}{\rho_1} \right)^{0.5} \left(\frac{\mu_1}{\mu_2} \right)^{0.1}. \quad (D.39)$$

and x is the equilibrium quality.

The S factor can be expressed as [D.14]

$$S = \left[1 + 0.12(Re_{tp})^{1.14} \right]^{-1} \quad \text{if } Re_{tp} < 32.5, \quad (D.40a)$$

or

$$S = \left[1 + 0.42(Re_{tp})^{0.78} \right]^{-1} \quad \text{if } 32.5 \leq Re_{tp} \leq 70.0, \quad (D.40b)$$

where

$$Re_{tp} = \frac{10^{-4} |v_2| \rho_2 (1 - \alpha)}{\mu_2} D_h F^{1.25}. \quad (D.41)$$

The above correlations for the suppression factor do not approach the correct limit (zero) as the void fraction approaches one. In the above equations, $S \rightarrow 1.0$ as $\alpha \rightarrow 1.0$. The following procedure is used to ensure that S approaches the correct limit which is zero. For $\alpha > 0.70$, S is calculated at $\alpha_s = 0.70$, and at the current value of α and the minimum value of the two suppression factors, S_{min} , is saved. Linear interpolation is then used between S_{min} and $S = 0$ at $\alpha = \alpha_c$,

$$S = \frac{S_{\min}(\alpha_c - \alpha)}{\alpha_c - \alpha_s}$$

In MELPROG/MOD1, $\alpha_c = 0.98$ and $\alpha_s = 0.70$. For $\alpha > \alpha_c$, $S = 0.0$.

The properties are evaluated at the liquid and vapor temperatures; x is the equilibrium quality, and v_2 is the liquid velocity parallel to the surface. Because the nucleate-boiling contribution to the Chen correlation was developed for saturated conditions [D.11], h_{nucb} is multiplied by a temperature ratio to adjust the HTC to the actual T_2 , Eq.(D.36). Because MELPROG/MOD1 can calculate superheated liquids, the adjustment factor is restricted to a maximum of 1.0 so that the adjustment is made for subcooled liquid only.

The vapor HTC goes from zero at $T_w = T_s$ to the transition-boiling value at $T_w = T_{CHF}$ (Section D.3.2.3); thus,

$$h_1 = \left(\frac{T_w - T_s}{T_{CHF} - T_s} \right)^2 \max(h_{nc}, h_{dr}) ,$$

where h_{nc} and h_{dr} are the natural-convection and Dougall-Rohsenow HTCs [D.17], respectively. The vapor HTC is calculated in subroutine HVFILM. For a void fraction greater than a cutoff value, α_c , linear interpolation is used between the current values of h_2 and h_1 and the values that are calculated for single-phase vapor; that is, $h_2 = 0$ and h_1 , calculated as discussed in Section D.3.2.5. This linear interpolation ensures that the boiling curve is smooth between heat-transfer regimes.

D.3.2.3 Transition-Boiling HTCs (Heat-Transfer Regime 3)

Transition boiling may be considered as a combination of nucleate and film boiling. A given spot on the wall surface is wet part of the time and dry during the remainder of the time. Therefore, contributions to the liquid and vapor HTCs exist for all conditions.

The total wall-to-fluid heat flux is obtained from a quadratic interpolation between the CHF and the minimum stable film-boiling points [D.14],

$$q_{trans} = \delta q_{CHF}'' + (1.0 - \delta) q_{min}'' , \quad (D.42)$$

where q''_{CHF} is the heat flux at CHF conditions, found from the Biasi correlation (Section D.3.3), and q''_{min} is the heat flux at the minimum stable film-boiling point, that is, the intersection of the transition- and the film-boiling points. This point is found from the homogeneous-nucleation correlation, as discussed in Section D.3.4. δ is a function of the wall temperature and the temperatures corresponding to q''_{CHF} and q''_{min} ,

$$\delta = \left(\frac{T_w - T_{min}}{T_{CHF} - T_{min}} \right)^2 \quad (D.43)$$

The vapor HTC is

$$h_1 = \max (h_{nc}, h_{DR}) ,$$

where h_{nc} and h_{DR} are the natural-convection and Dougall-Rohsenow HTCs [D.17], respectively. These correlations are discussed in the next section.

As in the nucleate-boiling heat-transfer regime, linear interpolation is used for $\alpha > \alpha_c$. The liquid HTC is

$$h_2 = \frac{q_{trans} - h_1(T_w - T_1)}{T_w - T_2} .$$

D.3.2.4 Film-Boiling HTCs (Heat-Transfer Regime 4)

In the film-boiling heat-transfer regime, radiative and dispersed-flow heat transfer occurs between the surface and the liquid; convective heat transfer occurs between the wall and the adjacent vapor. The liquid HTC is given by

$$h_2 = (h_r + h_{fb}) \left[\frac{T_w - T_s}{T_w - T_2} \right] + h_{df} ,$$

where h_r , h_{fb} and h_{df} account for radiation, film boiling and dispersed-flow.

The radiative contribution is

$$h_r = (1 - \alpha) \sigma \epsilon \left[\frac{T_w^4 - T_s^4}{T_w - T_s} \right] , \quad (D.44)$$

where σ is the Stefan-Boltzmann constant and ϵ is the wall

emissivity. In Eq.(D.44) the liquid absorptivity is 1.0.

The Bromley film-boiling HTC, h_{fb} , is given by [D.15]:

$$h_{fb} = 0.62 \left[\frac{\rho_1 k_1^3 (\rho_2 - \rho_1) g L'}{\mu_2 (T_w - T_s) \lambda} \right]^{1/4}, \quad (D.45)$$

where the characteristic length, λ , is

$$\lambda = 2\pi \left[\frac{\sigma}{g(\rho_2 - \rho_1)} \right]^{1/2}.$$

The latent heat of vaporization is modified as suggested in Ref. [D.20],

$$L' = L + 0.5 c_{p1} (T_1 - T_s).$$

The dispersed-flow HTC, h_{df} , uses the Forslund and Rohsenow equation [D.21], modified by multiplying $(1 - \alpha)$ by the entrained liquid fraction, E ; thus,

$$h_{df} = 0.2 c_1 \left[(1 - \alpha) E \right]^{0.6667} \text{BRAC}^{0.25} \left(\frac{T_w - T_s}{T_w - T_2} \right), \quad (D.46)$$

where c_1 is a constant equal to 1.2760 and

$$\text{BRAC} = \frac{g \rho_2 \rho_1 L' k^3}{|T_w - T_s| \mu_1 D_d}.$$

The dispersed-flow HTC is set equal to zero if $(1 - \alpha)E > 0.05$. Equation (D.46) is multiplied by the temperature ratio to change the base of the HTC from T_s to T_2 . The droplet diameter, D_d , is found from a Weber number criterion of 4.0:

$$D_d = \frac{We \sigma}{\rho_1 (v_1 - v_2)^2}.$$

The droplet diameter is restricted to the range:

$$1.0 \times 10^{-4} \leq D_d \leq 3.0 \times 10^{-3}.$$

The entrained liquid fraction is found in the following manner:

$$E = 0$$

$$\text{if } |v_1| \leq v_E ,$$

or

$$E = 1.0 - \exp 0.23 \left\{ - (|v_1| - v_E) \right\} \quad \text{if } |v_1| > v_E ,$$

where the entrainment velocity is:

$$v_E = 3.65 \left[\frac{(\rho_2 - \rho_1)\sigma}{\rho_1^2} \right]^{1/4}$$

and E is restricted to values between 0.07 and 1.0.

The vapor HTC is the maximum of the Bromley, natural-convection, and Dougall-Rohsenow values,

$$h_1 = \max (h_{nc}, h_{DR}) \quad . \quad (D.47)$$

The turbulent natural-convection equation used in this heat-transfer regime is [D.16]

$$h_{nc} = 0.13k_1 \left[\frac{\rho_1^2 g |T_w - T_1|}{\mu_1^2 T_1} \right]^{1/3} Pr_1^{1/3} \quad . \quad (D.48)$$

The forced-convection equation is based on Dougall and Rohsenow's modification [D.17] to the Dittus-Boelter equation [D.16],

$$h_{fc} = 0.023 \frac{k_1}{D_h} \left\{ \frac{\rho_1 [\alpha |v_1| + (1 - \alpha) |v_2|] D_h}{\mu_1} \right\}^{0.8} \left(\frac{\mu_1 c_{p1}}{k_1} \right)^{0.4} ,$$

where the Reynolds number is modified to reflect the volumetric flow rate of the two-phase mixture. As in the previous heat-transfer regimes, linear interpolation is used for $\alpha > \alpha_c$.

D.3.2.5 Single-Phase Vapor HTC (Heat-Transfer Regime 6)

For the single-phase vapor heat-transfer regime, $h_2 = 0$ and h_1 is the maximum of the turbulent natural convection and the Dittus-Boelter correlations, Eq.(D.48) and Eq.(D.25), respectively (these are now evaluated with the use of vapor properties).

D.3.2.6 Condensation HTC's (Heat-Transfer Regime 11)

Nusselt's analysis [D.2] is used to calculate the vapor HTC when condensation occurs. For an equilibrium quality $x \leq 0.71$, h_1 is set to zero. For $x > 0.71$, h_1 is found by linear interpolation between zero and the value of h_1 found from Nusselt's analysis. The liquid HTC is calculated using the Chen correlation [D.11]. The S factor is set equal to zero. The Chen correlation is based upon data taken up to $x = 0.71$. For $x > 0.71$, the Chen correlation is evaluated at $x = 0.71$, and h_2 is found by linear interpolation between the value of h_2 at $x = 0.71$ and the single-phase vapor h_2 of zero.

D.3.3 Critical Heat Flux (CHF)

The CHF point has two purposes in relation to the MELPROG/MOD1 boiling curve. First, the CHF point indicates the change from nucleate boiling (heat-transfer regime 2) to transition boiling (regime 3). Second, the CHF point is used in the quadratic interpolation that gives the transition-boiling liquid HTC. In TRAC [D.1], no CHF calculation is performed if $IC_{HF} = 0$ or a steady-state calculation is being performed ($ISTDY = 1$). In MELPROG, IC_{HF} is always 1 and the Biasi forced-flow CHF correlation [D.18] is used.

The Biasi correlation has a data base that covers the mass-flux range, G , between $100 \text{ kg/m}^2\text{s}$ and $6000 \text{ kg/m}^2\text{s}$. For mass-flux values between 0 and $200 \text{ kg/m}^2\text{s}$, the Biasi correlation is evaluated at $200 \text{ kg/m}^2\text{s}$. For negative values of G , the absolute value of G is used. The Biasi CHF correlation consists of two equations for q'_{CHF} , and the maximum CHF value calculated by the two equations is used,

$$q'_{CHF} = \frac{1.883 \times 10^7}{D_h^n G^{1/6}} \left[\frac{f_p}{G^{1/6}} - x \right] \quad (D.49)$$

and

$$q'_{CHF} = \frac{3.78 \times 10^7}{D_h^n G^{0.6}} h_p (1 - x) \quad , \quad (D.50)$$

where

$$n = 0.4, \quad \text{for } D_h \geq 1 \text{ cm};$$

$$n = 0.6, \quad \text{for } D_h < 1 \text{ cm};$$

$$f_p = 0.7249 + 0.099p \exp(-0.032p);$$

$$h_p = -1.159 + 0.149p \exp(-0.019p) + \frac{8.99p}{10 + p^2};$$

$$D_h = \text{hydraulic diameter (cm)};$$

$$G = \text{mass flux } \left(\frac{\text{g}}{\text{cm}^2 \text{ s}} \right);$$

$$p = \text{pressure (bar)}; \text{ and}$$

$$x = \text{equilibrium quality.}$$

Note that the Biasi correlation uses cgs units, but the constants in Eqs. (D.49) and (D.50) have been changed so that q'_{CHF} is in W/m^2 .

Predictions made with earlier versions of TRAC show that the Biasi correlation sometimes fails to predict CHF at high void fraction, even though the data indicate that CHF has occurred. To correct this problem, the Biasi correlation is used for a void fraction less than 0.97, and linear interpolation is used between the HTC at this void fraction and the one at 0.98, with the assumption that the T_{CHF} is 0.5 °K above T_s . For a void fraction greater than 0.98, the T_{CHF} is fixed at 0.5 °K above T_s .

Once q'_{CHF} is obtained, the temperature corresponding to the CHF point, T_{CHF} , is calculated by using a Newton-Raphson iteration to determine the intersection of the heat flux found by using the nucleate-boiling HTC and the CHF. An iteration is required because $T_w = T_{\text{CHF}}$ must be known to evaluate the Chen correlation; and, in turn, the Chen HTC must be known to calculate the wall temperature,

$$q'_{\text{CHF}} = h(T_w - T_s) \quad . \quad (D.51)$$

The equation for T_{CHF}^{n+1} is

$$T_{CHF}^{n+1} = T_{CHF}^n - \frac{\left(T_{CHF}^n - T_s - \frac{q_{CHF}'}{h} \right)}{\left[1 + \left(\frac{q_{CHF}'}{h^2} \frac{dh}{dT_w} \right) \right]}, \quad (D.52)$$

where T_{CHF}^n is the CHF temperature for the nth iteration, h is the HTC evaluated by using the Chen correlation, and dh/dT_w is the derivative of the HTC with respect to the wall temperature.

Convergence occurs if $|T_{CHF}^{n+1} - T_{CHF}^n| < 1.0$. A maximum of ten iterations is allowed; if convergence does not occur, a message is printed and a fatal error occurs.

The CHF temperature is restricted to the range, $T_s + 0.5 \leq T_{CHF} \leq T_s + 100$. The CHF calculations are done in subroutines CHF and CHF1.

D.3.4 Minimum Stable Film-Boiling Temperature, T_{min}

The minimum stable film-boiling point is the intersection point between the transition- and film-boiling heat-transfer regimes (Fig. D.1). In addition, this point is one of the points used in the interpolation scheme for the calculation of the transition-boiling heat flux.

The homogeneous-nucleation minimum stable film-boiling temperature correlation [D.19] is

$$T_{min} = T_{nh} + (T_{nh} - T_2)R^{0.5}, \quad (D.53)$$

where

$$R = \frac{(k\rho c)_2}{(k\rho c)_w} \quad (D.54)$$

and T_{nh} is the homogeneous-nucleation temperature. In Eq. (D.54) the subscript 2 indicates liquid properties and the subscript w indicates wall properties. The homogeneous-nucleation temperature is given by:

$$T_{nh} = 705.44 - (4.722 \times 10^{-2})DP + (2.3907 \times 10^{-5})DP^2 - (5.8193 \times 10^{-9})DP^3,$$

where $DP = 3203.6 - P$. The pressure P is in psia units, and T_{nh}

is in degrees Fahrenheit. In MELPROG, the variable P is converted to a temporary variable in British units, and T_{nh} is converted to Kelvin after the equation is evaluated.

After T_{min} is evaluated, the corresponding heat flux, q'_{min} , in the transition-boiling HTC interpolation can be calculated,

$$q'_{min} = h_{2min}(T_{min} - T_2) + h_{1min}(T_{min} - T_1) + h_{fb}(T_{min} - T_{sat}) ,$$

where h_{2min} and h_{1min} are the liquid and vapor HTCs, respectively, evaluated at the minimum stable film-boiling temperature. Thus,

$$h_{2min} = h_r \left(\frac{T_{min} - T_s}{T_{min} - T_2} \right) + h_{df}$$

and

$$h_{1min} = \max (h_{nc}, h_{DR}) .$$

If $\alpha > \alpha_c$, linear interpolation again is used between the values of the HTCs at α_c and the single-phase vapor values.

D.4 References

- D.1. TRAC-PF1/MOD1: An Advanced Best-Estimate Computer Program for Pressurized Water Reactor Thermal-Hydraulic Analysis, NUREG/CR-3858, LA-10157-MS, Los Alamos National Laboratory Los Alamos, NM (1986).
- D.2. J.G. Collier, Convective Boiling and Condensation (New York NY: McGraw-Hill Book Company, 1972).
- D.3. W.M. Rohsenow and H.Y. Choi, Heat, Mass, and Momentum Transfer, (Englewood Cliffs, NJ: Prentice-Hall Inc., 1961).
- D.4. S. Lekach, Development of a Computer Code for Thermal Hydraulics of Reactors (THOR), BNL-19978, Brookhaven National Laboratory Quarterly Progress Report (1975).
- D.5. C.J. Crowley, J.A. Block, and C.N. Cary, Downcomer Effects in a 1/15-Scale PWR Geometry: Experimental Data Report, NUREG-0281, Creare, Inc. report (May 1977).

- D.6. W.C. Rivard and M.D. Torrey, Numerical Calculation of Flashing from Long Pipes Using a Two-Field Model, LA-6104-MS, Los Alamos National Laboratory, Los Alamos, NM (November 1975).
- D.7. K. Lee and D.J. Ryley, "The Evaporation of Water Droplets in Superheated Steam," ASME paper 68-HT-11 (1968).
- D.8. G.W. Govier and K. Aziz, The Flow of Complex Mixtures in Pipes (New York, NY: Van Nostrand Reinhold Company, 1972).
- D.9. K. Kataoka and M. Ishii, Mechanism and Correlation of Droplet Entrainment and Deposition in Annular Two-Phase Flow, NUREG/CR-2885, ANL-82-44, Argonne National Laboratory, Argonne, IL (July 1982).
- D.10. G.B. Wallis, One-Dimensional Two-Phase Flow (New York, NY: McGraw-Hill Book Inc., 1969).
- D.11. J.C. Chen, "A Correlation for Boiling Heat Transfer of Saturated Fluids in Convective Flow," ASME paper 63-HT-34 (1963).
- D.12. L.S. Tong and J. Weisman, Thermal Analysis of Pressurized Water Reactors, Second Edition (LaGrange Park, IL: American Nuclear Society, 1979).
- D.13. J.P. Holman, Heat Transfer, Third Edition, McGraw-Hill Book Company, New York, NY (1972).
- D.14. T.A. Bjornard and P. Griffith, "PWR Blowdown Heat Transfer," Thermal and Hydraulic Aspects of Nuclear Reactor Safety 1, American Society of Mechanical Engineers, New York, NY, pp. 17-41 (1977).
- D.15. L.A. Bromley, "Heat Transfer in Stable Film Boiling," Chem. Eng. Prog. 46, pp. 221-227 (May 1950).
- D.16. W.H. McAdams, Heat Transmission, Third Edition (New York, NY: McGraw-Hill Book Company, 1954).
- D.17. R.S. Dougall and W.M. Rohsenow, Film Boiling on the Inside of Vertical Tubes with Upward Flow of the Fluid at Low Qualities, Massachusetts Institute of Technology Mechanical Engineering report 9079-26 (1963).

- D.18. L. Biasi, G C. Clerici, S. Garribba, R. Sala, and A. Tozzi, "Studies on Burnout: Part 3," Energ. Nucl., Milan, 14, pp. 530-536 (1967).
- D.19. R.E. Henry, "A Correlation for the Minimum Film Boiling Temperature," AIChE Symposium Series 138, pp. 81-90 (1974).
- D.20 S.S. Kutateladze, "Heat Transfer During Film Boiling," Heat Transfer in Condensation and Boiling, AEC-TR-3770, Atomic Energy Commission, Washington D.C. (1952).
- D.21 R.P. Forslund and W.M. Rohsenow, "Dispersed Flow Film Boiling," J. Heat Trans. 90, pp. 399-407 (1968).

U.S. Nuclear Regulatory Commission (28)
Office of Nuclear Regulatory Research
Division of Reactor System Safety
Washington, DC 20555

Attn: D. Ross	J. Han (5)	T. Lee
G. Marino	P. Worthington	B. Sheron
C. Kelber	R. Wright (5)	F. Costanzi
J. Mitchell	R. VanHouten	F. Eltawila
R. Meyer	B. Burson	N. Zuber
L. Chan	T. Walker	S. Fabic
F. Odar	L. Shotkin	

U.S. Nuclear Regulatory Commission
Office of Nuclear Reactor Regulation
Washington, DC 20555
Attn: R. Palla

U.S. Department of Energy
Office of Nuclear Safety Coordination
Washington, DC 20545
Attn: R. W. Barber

U.S. Department of Energy (2)
Albuquerque Operations Office
P. O. Box 5400
Albuquerque, NM 87185
Attn: J. R. Roeder, Director
Transportation Safeguards Division
J. A. Morley, Director
Energy Research Technology Division

Argonne National Laboratory
9700 South Cass Avenue
Argonne, IL 60439
Attn: W. Sha

Battelle Columbus Laboratory (2)
505 King Avenue
Columbus, OH 43201
Attn: R. Denning
C. Alexander

Battelle Northwest (3)
P. O. Box 999
Richland, WA 99352
Attn: F. Panisko
D. Lanning
N. Lombardo

Brookhaven National Laboratory (3)
Upton, NY 11973
Attn: R. A. Bari
T. Ginsberg
M. Khatib-Rhabar

Electric Power Research Institute
P. O. Box 10412
Palo Alto, CA 94303
Attn: R. Sehgal (2)

Los Alamos National Laboratory (12)
P. O. Box 1663
Los Alamos, NM 87545
Attn: J. Dearing
E. Fugelso
M. Sahota
H. Sullivan
J. Spore (5)
S. Woodruff
R. Jenks
F. Motley

Oak Ridge National Laboratory (3)
P. O. Box Y
Oak Ridge, TN 37830
Attn: T. Kress
S. Hodge
L. Ott

EG&G Idaho (4)
P. O. Box 1625
Idaho Falls, ID 83415
Attn: C. Allison
P. Bayless
J. Dahlman
P. Wheatley

Purdue University
Nuclear Engineering Building
West Lafayette, IN 47907
Attn: Professor M. Ishii

University of Wisconsin
Nuclear Engineering Department
1500 Johnson Drive
Madison, WI 53706
Attn: M. L. Corradini

H. Bairiot
Belgonucleaire S. A.
Rue de Champ de Mars 25
B-1050 Brussels
BELGIUM

D. Haas
Belgonucleaire S. A.
Rue de Champ de Mars 25
B-1050 Brussels
BELGIUM

Atomic Energy Canada Ltd. (3)
Chalk River, Ontario
CANADA KOJ 1J0
Attn: P. Fehrenbach
M. Notley
D. Nishimura

H. Rosinger
Atomic Energy Canada Ltd.
Pinawa, Manitoba
CANADA ROE 1LO

J. P. Longworth
Central Electric Generating Board
Berkeley Nuclear Labs
Berkeley, Gloucester GL13 9PB
ENGLAND

UK Atomic Energy Authority (2)
SRD
Culcheth, Warrington WA3 4VE
ENGLAND
Attn: M. Hayns
J. Stephenson

UK Atomic Energy Authority (3)
AEEW
Winfrith, Dorchester
Dorset DT2 8DH
ENGLAND
Attn: R. Potter
S. Kinnersley
J. Lillington

B. Turland
UK Atomic Energy Authority
Culham Laboratory
Oxon OX14 30B
Abingdon
ENGLAND

Kernforschungszentrum (4)
Postfach 3640
7500 Karlsruhe
FEDERAL REPUBLIC OF GERMANY
Attn: A. Fiege
S. Hagen
P. Hoffman
H. Rininsland

G. Petrangeli, Dire R. D. S.
Enea Nucl Energ Alt Disp
Via V Brancati 48
00144 Rome
ITALY

G. Saponaro
Enea Nucl Energ Alt Disp
Via Regina Marg-125
00144 Rome
ITALY

Japan Atomic Energy Res. Inst. (4)
Tokai-Mura Naka-Gun
Ibaraki-Ken 319-11
JAPAN
Attn: K. Hirano
S. Saito
K. Soda
K. Matsumoto

K. J. Brinkmann
Netherlands Energy Res. Fdtn.
P. O. Box 1
1755ZG Petten NH
NETHERLANDS

J. Bagues
Consejo de Seguridad Nuclear
Sor Angela de la Cruz 3
Madrid 28046
SPAIN

K. Johansson
Studsvik Energiteknik AB
S-611 82 Nykoping
SWEDEN

C. Graeslund
Statens Kernkraftinspektion
P. O. Box 27106
S-10252 Stockholm
SWEDEN

andia Distribution:

3141 S. A. Landenberger (5)
3151 W. I. Klein
6400 D. J. McCloskey
6415 R. M. Cranwell
6418 J. E. Kelly
6420 J. V. Walker
6422 D. A. Powers
6422 R. Elrick
6422 J. E. Brockmann
6423 K. O. Reil
6423 R. O. Gauntt
6423 B. W. Marshall, Jr.
6425 S. S. Dosanjh (20)
6425 R. D. Gasser
6425 A. J. Grimley, III
6425 T. J. Heames
6425 A. W. Reed
6425 R. C. Schmidt
6425 R. C. Smith
6425 J. L. Tomkins (4)
6425 M. F. Young
6427 M. Berman
6429 K. D. Bergeron
8524 J. A. Wackerly

NRC FORM 335 (2-84) NRCM 1102, 3201, 3202	BIBLIOGRAPHIC DATA SHEET	1. REPORT NUMBER (Assigned by TIDC, add Vol. No., if any) NUREG/CR-5193 SAND88-1824				
2. TITLE AND SUBTITLE MELPROG-PWR/MOD1: A Two-Dimensional, Mechanistic Code for Analysis of Reactor Core Melt Progression and Vessel Attack Under Severe Accident Conditions		3. LEAVE BLANK				
5. AUTHOR(S) S. S. Dosanjh, Ed.		4. DATE REPORT COMPLETED <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 50%; text-align: center;">MONTH</td> <td style="width: 50%; text-align: center;">YEAR</td> </tr> <tr> <td style="text-align: center;">January</td> <td style="text-align: center;">1989</td> </tr> </table>	MONTH	YEAR	January	1989
MONTH	YEAR					
January	1989					
7. PERFORMING ORGANIZATION NAME AND MAILING ADDRESS (Include Zip Code) Sandia National Laboratories Albuquerque, NM 87185		6. DATE REPORT ISSUED <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 50%; text-align: center;">MONTH</td> <td style="width: 50%; text-align: center;">YEAR</td> </tr> <tr> <td style="text-align: center;">May</td> <td style="text-align: center;">1989</td> </tr> </table>	MONTH	YEAR	May	1989
MONTH	YEAR					
May	1989					
10. SPONSORING ORGANIZATION NAME AND MAILING ADDRESS (Include Zip Code) Division of Systems Research Office of Nuclear Regulatory Research U.S. Nuclear Regulatory Commission Washington, DC 20555		8. PROJECT/TASK/WORK UNIT NUMBER 9. FIN OR GRANT NUMBER A1342				
12. SUPPLEMENTARY NOTES		11. TYPE OF REPORT Technical				
13. ABSTRACT (200 words or less) Since the Three Mile Island accident, increased emphasis has been placed on the analysis of severe reactor accidents because such accidents represent the dominant public risk from nuclear reactors. The U.S. Nuclear Regulatory Commission has made the development of mechanistic models for severe accident progression a major priority. The purpose of these models is to provide detailed, best-estimate, coupled analyses of all the major phenomena involved in the reactor vessel and coolant system in the course of the accident. To meet this objective, the MELPROG computer code is being developed. This report describes the two-dimensional, Pressurized Water Reactor (PWR) version of the MELPROG computer code, MELPROG/PWR-MOD1. The purpose of MELPROG is to provide for an accident sequence a description of (1) the state of the reactor core and surrounding in-vessel environment as a function of time, (2) the disposition of core materials (in particular, fission products) contained within the reactor coolant system boundary as a function of time, and (3) the state of the debris that enters the containment if the reactor vessel fails. MELPROG can also be coupled to the TRAC-PF1 RCS thermal-hydraulics code to provide an integrated analysis of the behavior of core, vessel, and reactor coolant systems during severe accidents. MELPROG treats core degradation and loss of geometry, debris formation, core melting, attack on supporting structures, slumping, melt/water interactions and vessel failure. Emphasis is also placed on the rates of hydrogen, steam and fission product formation, and transport to containment during the accident.						
14. DOCUMENT ANALYSIS - a. KEYWORDS/DESCRIPTORS Melt Progression, Severe Accidents		15. AVAILABILITY STATEMENT Unlimited				
b. IDENTIFIERS/OPEN-ENDED TERMS		16. SECURITY CLASSIFICATION (This page) Unclassified (This report) Unclassified				
17. NUMBER OF PAGES		18. PRICE				