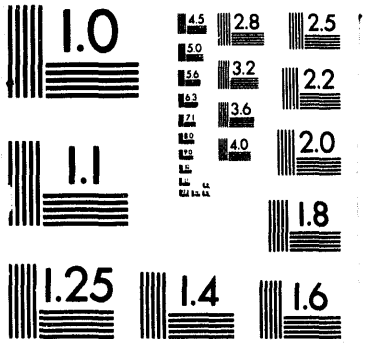

1.0
4.5
2.8
2.5
5.0
3.2
5.6
2.2
6.3
3.6
7.1
8.0
4.0
2.0
9.0
1.1
1.8
1.25
1.4
1.6

1 of 3

SAND92-0693
Unlimited Release
December 1993

Distribution
Category UC-~~40~~

SAFSIM Theory Manual – A Computer Program for the Engineering Simulation of Flow Systems

Dean Dobranich
Nuclear Technology Department
Sandia National Laboratories
Albuquerque, NM 87185

Abstract

SAFSIM (System Analysis Flow SIMulator) is a FORTRAN computer program for simulating the integrated performance of complex flow systems. SAFSIM provides sufficient versatility to allow the engineering simulation of almost any system, from a backyard sprinkler system to a clustered nuclear reactor propulsion system. In addition to versatility, speed and robustness are primary SAFSIM development goals. SAFSIM contains three basic physics modules: (1) a fluid mechanics module with flow network capability; (2) a structure heat transfer module with multiple convection and radiation exchange surface capability; and (3) a point reactor dynamics module with reactivity feedback and decay heat capability. Any or all of the physics modules can be implemented, as the problem dictates. SAFSIM can be used for compressible and incompressible, single-phase, multicomponent flow systems. Both the fluid mechanics and structure heat transfer modules employ a one-dimensional finite element modeling approach. This document contains a description of the theory incorporated in SAFSIM, including the governing equations, the numerical methods, and the overall system solution strategies.

Preface

This document is the first of three documents that describe the SAFSIM computer program. The second document (SAND92-0694) is an input manual that describes the input variables and format required to create an input file [illegible] SAFSIM execution. This first document is a theory manual that describes the governing equations and numerical methods of SAFSIM . The third document (SAND92-0695) is an application manual that provides various example problems, including benchmark problems. Originally, this theory manual was scheduled to be published first, followed by the input and application manuals. However, because of changing circumstances, the input manual was completed and published first. Because SAFSIM development is ongoing, the manuals should be considered living documents and therefore may not be up to date with the most current version of the computer program. Also, some of the features described in the theory manual may not be completely implemented in the program. Such features are noted in the description, but even these notes may be out of date, depending on progress made on program development. The application manual is in a very preliminary stage.

The development of SAFSIM was sponsored by the Air Force Phillips Laboratory under the Space Nuclear Thermal Propulsion program. The SAFSIM software is available to government agencies and their contractors through the Energy Science and Technology Software Center in Oak Ridge, TN.

Contents

List of Figures

List of Tables

Nomenclature

Fluid Mechanics

Variable	Description
A	superficial flow area
$\vec{A}$	control surface area vector
$A_{\min}$	minimum superficial flow area in a finite element
A_w	wall exchange surface area for convection heat transfer
a	speed of sound (sonic velocity)
B_i	fluid temperature gradient interpolation function for node i
B_M	manifold blowing parameter used for transpiration flow
B_0	manifold blowing parameter based on unmodified friction factor
Co	material Courant number ($v\Delta t/L$)
CL	control length
CS	control surface
CV	control volume
$[C]$	finite element fluid capacitance matrix
c_a	manifold element constant to account for annular geometry
c_d	choked flow boundary finite element discharge coefficient
c_p	fluid constant-pressure specific heat
$c_{p0} - c_{p3}$	four constants used to define specific heat polynomial function
c_v	fluid constant-volume specific heat
$c_1 - c_5$	five constants used to define porous media friction factors
D	equivalent diameter or particle diameter
D_h	hydraulic diameter
$[D_x^k]$	element mass fraction density matrix for fluid component k
d_m	coefficient matrix diagonal dominance factor
E_f	mechanical energy loss due to viscous forces
E_s	isentropic bulk modulus of elasticity
e	matrix entry
F	finite element fluidity (reciprocal of $\Re$)
$[F]$	fluidity matrix

f	friction factor
f_l	laminar flow friction factor
f_t	turbulent flow friction factor
$\bar{f}$	effective friction factor accounting for transpiration flow
G_i	finite element upwind interpolation function for node i
$\vec{g}$	acceleration vector
g_z	acceleration due to gravity in the vertical (z) direction
$\{g\}$	acceleration load vector
H_p	net pump head ($\Delta p/g\rho$)
h	fluid specific enthalpy
h_c	convection heat transfer coefficient
I_i	finite element conventional interpolation function for node i
J	super element index number
K	form loss coefficient (K factor)
$K_{1\text{-}2}$	form loss coefficient for flow from local node 1 to local node 2
$K_{2\text{-}1}$	form loss coefficient for flow from local node 2 to local node 1
$[K_A]$	finite element fluid advection conductance matrix
$[K_k]$	finite element fluid conduction conductance matrix
$[K_c]$	finite element fluid convection heat transfer conductance matrix
$[K_t]$	finite element total conductance matrix ($[K_k] + [K_c] + [K_A]$)
$[\bar{K}_t]$	effective finite element total conductance matrix
$[K_x^k]$	element mass fraction advective matrix for fluid component k
k_t	fluid thermal conductivity
$k_{t\xi}$	dispersion-enhanced fluid thermal conductivity (ξk_t)
$k_{t0} - k_{t3}$	four constants used to specify conductivity polynomial function
L	control volume or finite element flow length
$(L/D)_e$	equivalent length-to-diameter ratio to account for minor losses
M	Mach number (v/a)
$\{M_D\}$	dynamic mass flow rate load vector
$\{M_e\}$	effective mass flow rate load vector ($\{\dot{m}\}+\{S_e\}+\{g\}+\{M_D\}$)
m	fluid mass in a control volume or finite element (ρV)
$\dot{m}$	fluid mass flow rate ($\rho v \varepsilon A$)

$\dot{m}_R$	fluid mass flow rate after square-root relaxation
$\dot{m}_u$	updated fluid mass flow rate
$\dot{m}^*$	critical mass flow rate
$\{\dot{m}\}$	mass flow rate load vector
N_c	number of components in a multicomponent fluid
N_{eq}	number of equations
N_{ex}	number of exchange surfaces connected to a finite element
N_n	number of nodes in a finite element
N_S	number of elements in a super element
P	pump input power (power required to turn the shaft)
Pe_L	grid Peclet number (based on element length) ($Re_L Pr_\xi$)
Pr	Prandtl number ($\mu c_p / k_t$)
Pr_ξ	Prandtl number based on enhanced fluid conductivity ($\mu c_p / \xi k_t$)
P_h	heated perimeter
P_w	wetted perimeter
p	static or thermodynamic pressure
p_b	back pressure
$\bar{p}_L$	average pressure for a closed-loop network
p^*	critical pressure
$\{p\}$	node pressure vector
$\tilde{p}_i$	pseudo pressure at local node i ($p_i \chi_i / \rho_i$)
$\dot{Q}$	total rate of heat added to fluid
q	rate of heat added plus rate of shaft work done ($\dot{Q} + \dot{W}_{shaft}$)
q_ρ	effective power associated with density variations with time
q_d	power added directly to the fluid from any external source
q_{flo}	mechanical power converted to thermal power in a flowing fluid
q_k	power associated with conduction within the fluid
q_t'''	total volumetric heating rate to fluid
q_t	total heating rate to fluid
R	mixture gas constant (for a multicomponent fluid)
Re	Reynolds number based on superficial flow area ($\dot{m}D / A\mu$)
Re_L	grid superficial Reynolds number based on length ($\dot{m}L / A\mu$)

Re_l	superficial Reynolds number below which the flow is laminar
Re_u	superficial Reynolds number above which the flow is turbulent
R^k	gas constant for component k of a multicomponent fluid
$\{R_{q'''}\}$	finite element fluid volumetric heating load vector
$\{R_c\}$	finite element fluid convection heat transfer load vector
$\{R_t\}$	finite element total thermal load vector ($\{R_{q'''}\} + \{R_c\}$)
$\{\bar{R}_t\}$	effective finite element total thermal load vector
$\{R_x^k\}$	finite element mass fraction load vector for fluid component k
$\mathfrak{R}$	control volume or finite element effective flow resistance
S	mass source term for a fluid mixture
S_0	manifold sucking parameter
S^k	mass source term for fluid component k
St	Stanton number ($h_c/\rho v c_p$)
$\widetilde{St}$	modified Stanton number ($St \cdot A_w/\varepsilon A$)
$\{S_e\}$	effective fluid mass source load vector
s	pump rotational shaft speed (revolutions/s)
s_p	surface area of a particle in a porous media
T	fluid temperature
$\bar{T}_L$	average fluid temperature in a closed-loop network
T_R	fluid temperature after relaxation
T_u	updated fluid temperature
T_w	wall exchange surface temperature for convection heat transfer
T_0	constant used in the fluid viscosity equation
t	time
u	fluid specific internal energy
V	volume available for fluid in a finite element (εAL)
v_p	volume of particle in a porous media
v_p/s_p	volume-to-surface area ratio of particle in a porous media
W_i	finite element weighting function for node i
$\dot{W}$	rate of work done on fluid
w_f	friction factor weighting factor
w_c	log-mean fluid temperature weighting factor for convection term
w_d	log-mean fluid temperature weighting factor for dynamic term

w_m	mass flow rate or mass fraction weighting factor
$\dot{w}$	volumetric flow rate ($v\varepsilon A$)
x	position
x^k or x^l	mass fraction for component k (or l) of a multicomponent fluid
$\{x^k\}$	node mass fraction vector for fluid component k
z	vertical position

Greek Variables	**Description**
α	kinetic energy correction factor
α_p	closed-loop pressure correction factor
β_e	coefficient of thermal expansion
γ	ratio of fluid specific heats (c_p/c_v)
Γ	finite element boundary surface
Δp	finite element pressure drop or rise ($p_2 - p_1$)
$\Delta\hat{p}$	peak pump pressure rise
ΔT_{LM}	log-mean temperature difference
Δt_f	fluid mechanics time step
Δz	finite element change in elevation ($z_2 - z_1$)
ε	finite element porosity
ε_m	mass flow rate relative error
ε_T	fluid temperature relative error
ε_ρ	fluid density relative error
εA	finite element pore flow area
εv	superficial fluid velocity
$\widehat{\varepsilon v}$	maximum superficial fluid velocity
ϵ	absolute wall roughness
ϵ/D	relative wall roughness
η	upwind factor for the thermal energy equation
η_F	upwind factor for the mechanical energy equation
η_x	upwind factor for the fluid mass fraction equation
θ_{FM}	implicitness factor for the fluid mechanics equations
φ	empirical correction factor for blowing flow correlation
λ	volumetric flow rate fraction

λ_m	relaxation parameter for mass flow rate iterations
λ_T	relaxation parameter for temperature iterations
λ_ρ	relaxation parameter for density iterations
$\Lambda_1 - \Lambda_3$	three constants used to specify fluid density polynomial function
μ	fluid dynamic viscosity
μ_0	constant used in the fluid viscosity equation
μ_w	dynamic viscosity based on the fluid temperature at the wall
v	pore fluid velocity
$v_\perp$	fluid velocity perpendicular to a manifold element wall
$\vec{v}$	control volume fluid velocity vector
ξ	fluid conductivity enhancement factor (for flow in porous media)
ρ	fluid density
ρ_R	fluid density after relaxation
ρ_u	updated fluid density
$(\rho R)_L$	gas density times the gas constant for a closed-loop network
$\vec{\tau}$	shear stress vector
ϕ_p	fluid dispersion conductivity coefficient
χ	mass flow rate fraction
ψ	constant used in the fluid viscosity equation
Ω	solution domain which is equivalent to the finite element volume

Subscripts	Description
1	port 1 of a control volume or local node 1 of a finite element
2	port 2 of a control volume or local node 2 of a finite element
3	port 3 of a manifold finite element (feed line connection)
i	node index number or matrix row index number
J	super element index number
j	finite element index number or matrix column index number
p	evaluated at constant pressure
r	rated condition
$shaft$	shaft
T	evaluated at constant temperature
v	evaluated at constant volume

vis	viscous
( )	indicates element number

Superscripts (and Overstrikes)	Description
n	current time (beginning of time step)
n+1	future time (end of time step)
θ_{FM}	evaluated at this time level
^	peak or maximum
~	pseudo or modified
—	average or effective
·	time rate of change
*	critical conditions
→	vector
'''	volumetric

Other Symbols	Description
{ }	column vector
⌊ ⌋	row vector
[ ]	matrix
| |	absolute value
Σ	summation
Δ	difference
∂	partial derivative
∫	integral
1-D	one dimension or one dimensional
2-D	two dimensions or two dimensional

Structure Heat Transfer

Variable	Description
A_{ex}	exchange surface area (P_hL_h)
A_k	structure finite element conduction area
$[C_s]$	structure finite element capacitance matrix
c	structure material specific heat

D	particle diameter for a porous media
e_r	surface emissivity for thermal radiation heat transfer
f_I	fraction of explicit time step limit for a structure
H	height of a cylindrical heat transfer structure
h_c	convection heat transfer coefficient for an exchange surface
$[K_c]$	structure finite element convection conductance matrix
$[K_k]$	structure finite element conduction conductance matrix
$[K_t]$	finite element total conductance matrix ($[K_k] + [K_c]$)
$[\overline{K}_t]$	effective finite element total conductance matrix
k_{PM}	porous media effective thermal conductivity
k_r	effective thermal radiation conductivity for a porous media
k_S	thermal conductivity of the solid structure in a porous media
k_s	structure material thermal conductivity
L_k	structure finite element conduction length
N_{ex}	number of convection exchange surfaces for a structure element
P_h	heat perimeter of an exchange surface
q	heat flow (power)
q''_c	convection heat flux
q'''_s	structure volumetric heat source
$\{R_c\}$	finite element convection heat transfer load vector
$\left\{R_{q''}\right\}$	finite element heat flux load vector
$\left\{R_{q'''}\right\}$	finite element volumetric heating load vector
$\{R_t\}$	finite element total thermal load vector ($\{R_{q'''}\} + \{R_{q''}\} + \{R_c\}$)
$\left\{\overline{R}_t\right\}$	effective finite element total thermal load vector
r	radial position
T	fluid exchange temperature for convection heat transfer
T_s	structure temperature
$\{T_s\}$	structure node temperature vector
t	time
V_s	structure finite element volume (A_kL_k)
W_i	finite element weighting function for node i
x	position

Greek Variables	Description
ΔT_s	finite element structure temperature difference ($T_2 - T_1$)
Δt_E	explicit time step limit
Δt_s	sub-time step for a heat transfer structure
ε	porosity
θ	meridonal angle
θ_{HT}	implicitness factor for the structure heat transfer equations
λ_s	dimensionless time step ($\Delta t_s / \Delta t_E$)
π	circumference of a circle divided by its diameter (3.14159...)
ρ_s	structure material density
σ_B	Stefan-Boltzmann constant ($5.6696 \cdot 10^{-8}$ W/m^{2}K^4)
ϕ	azimuthal angle
Ω	structure temperature solution domain (volume)

Subscripts	Description
1	local node 1 of a finite element
2	local node 2 of a finite element
BC	boundary condition
c	convection
E	explicit
ex	exchange surface
i	node index number
j	finite element index number
S	solid
s	structure

Superscripts (and Overstrikes)	Description
min	minimum value
n	current time (beginning of time step)
n+1	future time (end of time step)
—	effective
$''$	flux (per unit area)
$'''$	volumetric

Other Symbols	Description
$\{\ \}$	column vector
$\lfloor\ \rfloor$	row vector
$[\ ]$	matrix
Σ	summation
Δ	difference
∂	partial derivative
$\int$	integral
1-D	one dimension or one dimensional
2-D	two dimensions or two dimensional
3-D	three dimensions or three dimensional

Reactor Dynamics

Variable	Description
C_i	delayed neutron precursor concentration for group i
$d\rho$	total differential feedback reactivity
$d\rho_k$	differential feedback reactivity for feedback term k
$\frac{d\rho_p}{dt}$	time derivative of the programmed reactivity
H_j	decay heat precursor concentration for group j
k	feedback term index number
k_e	effective neutron multiplication factor
l	effective neutron lifetime
l_g	neutron generation time (l/k_e)
N_f	number of differential feedback terms
N_h	number of decay heat precursor groups
N_n	number of delayed neutron precursor groups
n	total or instantaneous neutron power or neutron density
n_e	effective thermal power ($n_p + n_h$)
n_h	decay heat thermal power
n_p	prompt neutron or fission thermal power
$\overline{n}$	time-averaged neutron power

q_n	extraneous neutron source
t	time
x_k	a system parameter for feedback reactivity term k

Greek Variables	Description
α_k	feedback coefficient for feedback term k
β	total delayed neutron fraction
β^H	total decay heat fraction
β_j^H	decay heat fraction for group j
β_i	delayed neutron fraction for group i
$\bar{\beta}$	effective total delayed neutron fraction ($\gamma\beta$)
γ	energy correction factor
Δt_r	sub-time step for reactor dynamics
Δx_k	the change in the feedback parameter x_k
ε_{C_i}	truncation error for delayed neutron concentration for group i
ε_n	relative truncation error for neutron power
ε_r	maximum relative truncation error
ε_r^d	desired relative truncation error
λ_i	decay constant for delayed neutron group i
λ_j^H	decay constant for decay heat group j
ρ	total reactivity
$\rho_\$$	reactivity in units of dollars ($) and cents (¢)
τ	reactor period
ω	inverse reactor period

Subscripts	Description
1	indicates solution at full time step
1/2	indicates solution at one-half time step
4*th*	indicates fourth-order solution
5*th*	indicates fifth-order solution
i	delayed neutron group i index number
j	decay heat group j index number
k	feedback term index number

r	reactor dynamics
$	dollars

Superscripts (and Overstrikes)	Description
0	initial value
H	decay heat
m	current time (beginning of time step)
$m+1$	future time (end of time step)
—	average or effective

Other Symbols	Description
ln	natural log
max	maximum value
Δ	difference
Σ	summation
$\vert\ \vert$	absolute value

1.0 Introduction

SAFSIM[✦] (System Analysis Flow SIMulator) is a FORTRAN computer program that provides engineering simulations of user-specified flow networks at the system level. It includes fluid mechanics, structure heat transfer, and reactor dynamics capabilities. SAFSIM provides sufficient versatility to allow the simulation of almost any flow system, from a backyard sprinkler system to a clustered nuclear reactor propulsion system. Although versatility is the primary SAFSIM development goal, computational speed and robustness are additional development goals. In general, these are conflicting goals. The enhancement of versatility usually results in a reduction in computational speed and sometimes in a reduction of robustness. An attempt is made to balance these conflicting goals; however, versatility received top priority. The next two subsections attempt to convey the motivation for SAFSIM development along with a discussion of the general guiding philosophy.

1.1 SAFSIM Development Motivation

The key feature of a system level computer program for flow modeling is the ability to simulate the integrated performance of all components of a system. Thus all the potentially complex interactions and feedback mechanisms between components can be captured. Also, in the simulation of the integrated performance of a system, the details of a component-level response are of less concern than the overall system response. In such simulations it is usually sufficient to model the components in a one-dimensional (1-D) or lumped fashion. Such system level programs have been developed and used with great success in the light-water nuclear reactor industry for many years (for example, Reference 1). However, such computer programs contain a significant amount of capability particular to light-water reactors and therefore are not truly general purpose. Also, making modifications to these programs is difficult without intimate familiarity of the program's overall structure.

A primary motivation for SAFSIM development is the desire to have a general-purpose computational tool to provide quick and inexpensive engineering performance simulations of complicated user-specified systems. The simulations are intended to provide a first-look understanding of a system's steady-state and transient behavior during operational and off-normal conditions. It is also desired that this tool have the ability to accommodate changes in the problem definition via changes in an input file rather than changes in the computer program. Thus all geometric and operational information is provided by the analyst, allowing the analyst to select a level of modeling detail consistent with the problem detail available. SAFSIM's network approach allows

[✦] pronounced *safe sim*.

the user to "build" a geometric representation of any system by appropriately coupling networks of fluid flow and structure heat transfer finite elements.

As mentioned, SAFSIM is a general-purpose computational tool. Thus it was designed to be independent of any specific system hardware, allowing it to be used for a variety of engineering applications. Although SAFSIM was originally intended to allow simulation of various advanced nuclear thermal propulsion concepts, hardware-specific capabilities were not incorporated. Instead, the basic physics relevant to modeling such systems was incorporated at a fundamental level. In addition, various applications of the basic physics require different modeling assumptions with regard to their implementation. For example, modeling the flow through a porous media depends on the specific correlations selected for friction factors and heat transfer coefficients. Also, the level of detail included in the input model can have significant impact on the calculated results. Therefore, SAFSIM was designed such that these modeling decisions are delegated to the analyst whenever practical, via the input file and several user-supplied subroutine interfaces. This approach allows the analyst to build an input model for a particular system and to then parametrically explore the impact of the various modeling assumptions. Such an approach generally results in a more thorough understanding of the system, although it requires more effort by the analyst. This general-purpose feature of SAFSIM was a key motivation in its development.

1.2 An Engineering Approach

SAFSIM is referred to as an engineering computer program. This engineering approach forms the basis of the general development philosophy adopted for SAFSIM. This subsection is intended to explain what is meant by an engineering approach, and to convey how this basic philosophy impacts SAFSIM development and its use. This subsection is comprised of a collection of thoughts, accumulated over a number of years by the author, with regard to engineering analyses. Such thoughts are, in general, very subjective; they are offered simply to provide the basis for SAFSIM development and to hopefully entice other analysts to examine their analysis requirements in a different light.

With regard to an analysis, an engineer should never ask the question: Are these results correct? Instead, the appropriate question to ask is: Are these results good enough? And "good enough" must be evaluated in the context for which the results are to be used. "Good enough" must also be consistent with the level of problem definition. Often in practice, the "correct" results are never known. Even the best-planned experiments contain uncertainties with respect to measurement and conduct of operation. Sometimes the exact configuration or condition of a system is not known with certainty.

For example, how accurately a specific component temperature must be predicted depends on the design and nature of the component. If the component was designed with a large margin of safety with respect to thermal stress, knowing its operating temperature within ±25% may be sufficient. However, if it is anticipated that the component will be operating close to its melting point, it may be necessary to predict its temperature to within ±5%. If the modeled component is part of a conceptual system (a "paper study"), general trends in the temperature response may be sufficient with little emphasis placed on the magnitude.

Multidimensional fluid flow and heat transfer computer programs abound and can be used to simulate the details within a single component. But in an engineering system level simulation, capturing the overall system response is of primary concern. Also, the complete equations that govern fluid mechanics are quite formidable and some level of approximation is required to achieve their solution in a reasonable amount of time. Thus the use of "engineering factors" to account for some of the details allows considerable reduction in computation time. An example of such an engineering factor is the form loss coefficient, also known as a *K*-factor. A *K*-factor can be used to account for the pressure losses associated with multidimensional flow disturbances occurring within a component. The recirculating flow occurring at an abrupt expansion in a pipe is an example of such a disturbance. At the system level, the details of such a disturbance are not important, only the pressure loss associated with it.

Other examples of engineering factors include friction factors, heat transfer coefficients (Newton's law of cooling), shape factors, effective material conductivities, fluid conductivity enhancement factors, hydraulic and equivalent diameters, and pump and turbine characteristic curves. Over the years, engineers have developed a wealth of engineering factors to simplify complex problems. Some of these factors are based on basic theory while others are based on experiment. SAFSIM takes advantage of as many of these engineering factors as possible. The analyst has access to these factors through many options in the input file and through user-supplied subroutine interfaces within SAFSIM. This is consistent with the SAFSIM development goal of versatility.

The impact on the use of engineering factors, along with the use of simplifications to the governing equations, can and should be investigated parametrically. Whenever calculated results are found to be sensitive to an assumption or simplification, additional detailed computational and experimental investigations would be warranted. This provides the impetus to allow the analyst as much control over the system model and solution procedure as practical. It has been stated that "If you give the user enough rope, he will surely hang himself." Unfortunately, when an analysis runs amok, usually the hangee blames the computer program. Such is the danger of providing so much rope. However, it is the author's observation that a good analyst appreciates lots of rope and simply accepts the danger of working with a noose around his neck.

This brings up the next sequence of thoughts. It is the responsibility of the analyst to determine if the calculated results are reasonable. Too much faith should never be placed in any computer program. The analyst must closely evaluate the results, perform parametric studies (possibly with other programs), and compare results with any available experimental data. SAFSIM is not intended to be used as a "black box."

As just stated, SAFSIM results should be compared to experimental data whenever they are available. However, "numerical experiments" provide a luxury not available to the hardware experimentalist. Computer simulation allows an analyst the ability to alter physics in order to understand the sensitivity of the results to a particular assumption. For example, the pressure drop associated with transpiration drag can be "turned off." Or exotic fluids and materials can be evaluated with very little expense. Because the user has complete control of the input models for SAFSIM, the analyst can perform "what-if analyses" to improve understanding of performance and to provide insight as to where additional detailed computational or experimental work should be focused.

SAFSIM is a powerful tool and if used correctly can provide approximate solutions to complex system level flow problems. However, it has limits and these limits should be clearly understood. This theory manual, along with the cited references, should be studied to ensure that SAFSIM is not being used to solve a problem it was never intended to solve. Like any computer program, SAFSIM can make a good analyst better or a bad analyst dangerous. Be wary of software advertising "no experience required."

The next section provides an overview of the SAFSIM computer program. It includes a qualitative description of its major features without any equations. This section is followed by the development of the fluid mechanics, structure heat transfer, and reactor dynamics equations and numerical methods. These sections contain many equations to allow scrutiny of the theory behind SAFSIM. Next comes a description of the overall system level solution procedure. This document concludes with some general summary statements followed by some appendices, which contain useful information pertaining to SAFSIM. Variables are defined after they are first used. In addition, the nomenclature section at the beginning of this document provides a quick reference for the definitions of all variables.

2.0 SAFSIM Overview

SAFSIM currently contains three basic physics modules: (1) fluid mechanics, (2) structure heat transfer, and (3) reactor dynamics. All three modules are coupled to allow the prediction of system performance. The analyst can employ any or all of the physics modules as the problem dictates. The following subsections provide a qualitative overview of the three physics modules along with a brief description of other SAFSIM features.

2.1 Fluid Mechanics

The fluid mechanics module solves the single-phase conservation of mass, momentum, and energy equations and is based on a 1-D finite element model. The module allows the specification of a user-defined flow network or multiple networks. The network approach allows finite elements to be connected in series or parallel; thus pseudo multidimensional representations of components can be constructed. The fluid can be either a gas or a liquid; also, mixing models are included that allow the specification of multiple fluids. Either open or closed networks can be modeled. The fluid mechanics module includes the solution of the compressible or incompressible mechanical and thermal energy equations.

The analyst can select a quasi-static or partially dynamic solution of the fluid mechanics equations. For the quasi-static solution, only the boundary conditions and loads (such as convection heat transfer from a structure) are time dependent; the fluid temperature, mass flow rate, and density time-derivative terms are neglected. For the partially dynamic solution option, the fluid temperature and mass flow rate time-derivative terms are included. A fully dynamic option is planned in which the density time-derivative term (mass accumulation) in the mass continuity equation is included. The dynamic options are fully implicit. Also, to speed execution, an option exists to bypass the mechanical energy equation solution during problem execution if desired. Although the mechanical and thermal energy equations are solved separately, they are iteratively coupled to provide the solution to a total energy equation for the fluid. Super element capability is provided for the mechanical energy equation solution. Super elements allow a series of finite elements to be grouped into one equivalent super element, greatly improving computational efficiency.

Advection✦ of, conduction in, and convection to the fluid are modeled in the thermal energy equation. Upwind finite elements are employed for the thermal energy equation with automatic determination of the upwind factor based on the

✦ The term "advection" refers to the movement of energy associated with a flowing fluid; it is used in place of the often-used term "convection" which is reserved for reference to convection heat transfer based on Newton's law of cooling.

grid Peclet number. Convective heat transfer from any solid structure to the fluid is based on the log-mean temperature difference, thus improving accuracy and allowing a coarser mesh. More than one convective exchange surface can be coupled to any fluid mechanics finite element.

Flow in porous media can be simulated. The input data for a fluid mechanics finite element allows the user to specify an element porosity to account for porous media. The analyst may select from among several popular correlations for calculation of the porous media friction factor. If none of these correlations is adequate, analysts can define their own via the input file. Fluid conductivity enhancement due to dispersive effects within the porous media is included.

An ideal equation of state is used for gases to specify density as a function of temperature and pressure. For liquids, the density is specified as a function of temperature only. Fluid conductivity and specific heat are specified by third-order polynomial equations. Either a power-law or Sutherland-law equation can be used to specify fluid viscosity. Multiple temperature ranges can be specified for the three fluid properties. A user-supplied equation of state and property interface is also available.

Different time steps can be specified for each fluid mechanics flow network. Pressure or mass flow rate boundary conditions can be applied to any node of a flow network for the mechanical energy equation. Temperature or zero heat flux boundary conditions can be applied to any node for the thermal energy equation. Mass fraction boundary conditions for each component of a multicomponent fluid network can also be specified for any node. A special closed-loop pressure boundary condition is available for simulating closed-loop pressure performance when using SAFSIM's quasi-static solution procedure.

Three special-purpose finite elements are included to increase modeling versatility: (1) a choked flow boundary element, (2) a compressor/pump element, and (3) a distributed flow manifold element. A choked flow boundary element allows the simulation of choked flow based on the quasi-static solution of the compressible, isentropic flow equations. Discharge coefficients can be added to this finite element to account for any flow inefficiencies. The compressor/pump finite element requires the input of a pump characteristic curve along with rated and operational speeds. The pump characteristic curve is adjusted for varying operational speeds based on the pump similarity laws. A distributed flow manifold finite element provides a quasi two-dimensional flow capability and allows flow to enter or exit this element along its entire length and not just at the nodes. It accounts for the merging of flow streams. Also, transpiration flow effects on the friction factor and heat transfer coefficients can be included in this element if desired. A fourth special-purpose finite element is planned that will allow simulation of a turbine.

Three equation solvers are available for the mechanical energy equation: (1) Gauss-Seidel (iterative), (2) Cholesky decomposition (direct), (3) and Gauss elimination (direct). The analyst may select any of these solvers based on the specific problem, if a specific solver is not selected, SAFSIM attempts to use Gauss-Seidel iteration first. If convergence is not met, or if the coefficient matrix is not diagonally dominant, Cholesky decomposition is employed. This is a direct solver that operates only on terms within the semibandwidth of the coefficient matrix and is therefore relatively fast. If the coefficient matrix is not positive definite, Cholesky will fail and Gauss elimination is used. This is the slowest of the three solvers but the most general. This solver uses partial pivoting and is written so that only the terms within the bandwidth of the coefficient matrix are included, greatly increasing speed. The use of multiple numerical solvers adds to the robustness of the program.

Two solvers are available for solution of the thermal energy equation: (1) Gauss-Seidel and (2) Gauss elimination. For flow problems that are advectively dominated (such as many gas flow problems), Gauss-Seidel provides a rapid solution for the fluid temperature field. Gauss elimination is included for flow problems that are not advectively dominated because the coefficient matrix may lose its diagonal dominance. Cholesky decomposition is not included for the thermal energy solution because the coefficient matrix is not symmetric due to the advective term in the equation.

Any computer program designed to solve the fluid mechanics equations requires decisions as to what aspects of fluid mechanics are to be included. These decisions determine what type of fluid flow problems can be simulated. The following table summarizes these decisions with those applicable to SAFSIM marked with a check, ✓. This table should provide prospective SAFSIM users with a quick reference for determining if SAFSIM is the appropriate computational tool.

Table 1. Fluid Mechanics Flow Types

Eulerian ✓	Lagrangian
continuum ✓	discrete
internal and porous media ✓	external
inviscid	viscous ✓
laminar ✓	turbulent ✓
incompressible ✓	compressible ✓
subsonic/sonic ✓	supersonic (✓anticipated)
Newtonian fluid ✓	non-Newtonian fluid
static ✓	dynamic ✓
one dimensional ✓	multidimensional
single phase ✓	multiphase
single-component fluid ✓	multicomponent fluid ✓

Note: a check mark (✓) indicates a flow type modeled in SAFSIM

2.2 Structure Heat Transfer

The structure heat transfer physics module is based on a 1-D finite element model that allows the user to model conduction in any heat transfer structure such as pipe walls, fuel elements or particles, vessel walls, thermocouples, etc. The user can include as many structures as desired. Multiple exchange surfaces allow the user to convectively or radiatively couple any heat transfer structure finite element to any fluid mechanics finite element(s). The module can be run in quasi-static or dynamic mode. In dynamic mode, the module automatically determines the optimum implicitness factor for each element of each structure at each time step. Time step size can be automatically calculated for each structure by the program or user specified.

Material properties can be temperature dependent if desired. Also, several options are available for supplying the property data, including tables, polynomials, power laws, and constants. An extensive built-in heat transfer coefficient correlation library includes correlations for laminar and turbulent flows, for internal and external flow geometries, and for gases and liquids (including liquid metals). Temperature, heat flux, and convective/radiative boundary conditions can be specified. Also, finite elements from different heat transfer structures can be conductively (including a contact resistance) or radiatively (including view factors) coupled. This coupling, however, is explicit. A tridiagonal numerical solver provides a rapid solution for the node temperatures of all of the structures.

2.3 Reactor Dynamics

The reactor dynamics physics module is based on a point (0-D) kinetics model with feedback. Multiple reactors can be specified, and multiple feedback coefficients are allowed for each reactor to account for all system interactions. The analyst has complete control over how the feedback coefficients are defined. Also, several "control laws" are available to simulate control rod/drum reactivity control for reactor startup and shutdown simulations. Special-purpose control laws can be added to the program by the analyst if desired. The analyst can specify any number of delayed neutron groups and any number of decay heat groups. A source term also can be included.

Two solvers are currently available for integration of the reactor dynamics equations: (1) Euler, and (2) Runge-Kutta-Fehlberg (RKF). Adaptive time stepping is employed by both solvers based on the desired relative truncation error. The Euler integrator employs step doubling to provide the truncation error estimate for time step selection. The RKF integrator determines an error estimate as the difference between a fourth- and fifth-order prediction. The analyst can switch between solvers during a problem if desired.

2.4 Function-Controlled Variables

In addition to the three physics modules, SAFSIM contains a special input feature: function-controlled variables. This feature allows the analyst to specify many of the input variables as functions of output variables. Examples of input variables include: convergence criteria, flow areas, conduction lengths, feedback coefficients, compressor speed, and boundary conditions. Examples of output variables include pressure, temperature, velocity, and heat transfer coefficient. For example, the diameter of a fluid mechanics finite element can be specified as a function of a heat transfer finite element temperature to simulate the effect of thermal expansion on the flow field. Functions can be specified as functions of other functions to allow complex logic paths to be included via input. A library of functions is included along with a provision for analysts to easily add their own. Examples of mathematical functions in the library include cosine, square root, averaging, table lookup, polynomials, and conditional statements.

2.5 System Level Features

Although SAFSIM can be used to model specific components in detail, its major strength is the ability to couple multiple components together to investigate synergistic effects between components. This is important because, in general, a system of optimized components does not produce an optimum system. Nonlinearities in the physics can produce system performance that might not be expected based on analysis of an isolated component.

The three physics modules, along with the function-controlled variables, are explicitly coupled. Employing explicit coupling greatly increases program versatility and allows the specification of different time steps for each fluid mechanics network, heat transfer structure, and nuclear reactor in the system, which all may have vastly different characteristic time constants. For very tightly coupled systems, the system level time step can be decreased as appropriate.

Although SAFSIM is a time-dependent computer program, it can be used to perform steady-state calculations of a system. Two methods are available to accomplish this. The first method involves executing a transient simulation until the time-derivative terms are sufficiently small. A second system level steady-state method sets all time-derivative terms to zero and then performs wall temperature iterations to obtain consistency between the physics modules and function-controlled variables. In general, this second method is very fast compared to the first method but for very complex systems may not be as robust. The two methods can be combined if desired.

In addition to the many options available via the input file, several user-supplied subroutine interfaces are supplied to allow an analyst to tailor SAFSIM to problem-specific modeling needs. These interfaces streamline the process of

adding special features and include user-supplied equation of state and fluid properties; user-supplied heat transfer coefficients; user-supplied functions; user-supplied reactor dynamics control laws; user-supplied loss-coefficients for manifolds, tees and wyes; user-supplied source terms; user-supplied system-level subroutine for additional physics modules or for coupling other programs to SAFSIM; and user-supplied special-purpose input and output.

2.6 Proposed Enhancements

SAFSIM is a functional computer program that runs on a personal computer or workstation and provides the analyst with a tool for obtaining engineering solutions to complex system analysis problems. SAFSIM development is by no means complete. However, the program architecture was designed with expansion and modification in mind; thus new features and capabilities can be added with relative programming ease. Also, SAFSIM was designed with many user interfaces built into the program, allowing each analyst to tailor SAFSIM for their particular application without having to fully understand the internal programming structure and solution techniques. Hopefully it is apparent that versatility is the primary development objective of SAFSIM, making it a candidate analysis tool for a broad range of engineering disciplines. Benchmarking and documentation of the current version are in progress. Additional enhancements are envisioned to make SAFSIM even more versatile, robust, and fast. These planned enhancements will expand the class of problems for which SAFSIM is applicable and are provided in the following nonprioritized list:

(1) LU decomposition solver with iterative refinement for very large flow networks (greater than about 350 nodes)
(2) Built-in bandwidth minimizer for the mechanical and thermal energy equation solvers
(3) Turbine finite element
(4) Kagonove solver for the reactor dynamics equations
(5) Structural mechanics physics module
(6) Restart capability
(7) Blowdown tank boundary condition option
(8) Generalized mass source terms for the fluid mechanics equations
(9) 2-D tables and other special functions
(10) Mass accumulation term in the continuity equation
(11) Upwind finite elements for the mechanical energy equation
(12) Liquid metal electromagnetic pump option
(13) Liquid metal accumulator model
(14) Homogeneous equilibrium model for two-phase flow
(15) Pre- and post-processing tools (graphical interface)

3.0 Fluid Mechanics Governing Equations

There are three equations governing fluid mechanics: (1) conservation of mass, (2) conservation of momentum, and (3) conservation of energy. These three equations are developed using a control volume formulation in which integral balance equations for the three conservation quantities (mass, momentum, and energy) serve as the starting point. An example, one-dimensional (1-D), arbitrary-geometry, fixed control volume with two ports for entering and exiting flow is provided in Figure 1. The integral balance equations and the details of their formulation can be found in most introductory text books on fluid mechanics. The following are recommended examples of such texts: *Introduction to Fluid Mechanics,*[2] *Fluid Mechanics,*[3] *Incompressible Flow,*[4] and *Transport Phenomena.*[5]

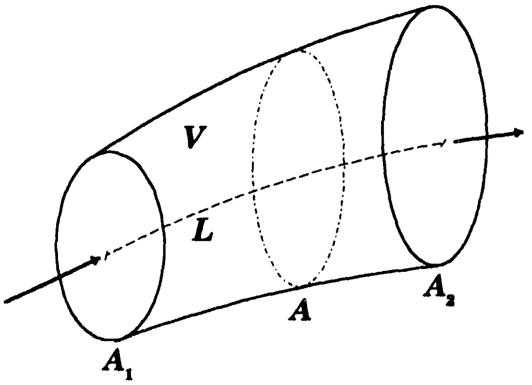


Figure 1. Example 1-D, Fixed Control Volume

The conservation of mass, momentum (in terms of kinetic energy), and energy integral balance equations for a fixed control volume are

$$\frac{\partial}{\partial t}\int_{CV}\rho dV + \int_{CS}\varepsilon\rho\vec{v}\cdot d\vec{A} = S \tag{3.1}$$

$$\frac{\partial}{\partial t}\int_{CV}\tfrac{1}{2}\rho v^2 dV + \int_{CS}\tfrac{1}{2}\varepsilon\rho v^2\vec{v}\cdot d\vec{A} = \int_{CV}\rho\vec{v}\cdot\vec{g}dV - \int_{CV}\vec{v}\cdot\nabla p dV - \dot{m}E_f \tag{3.2}$$

$$\frac{\partial}{\partial t}\int_{CV}\rho\left(u+\tfrac{1}{2}v^2+g_z z\right)dV + \int_{CS}\varepsilon\rho\left(u+\tfrac{1}{2}v^2+g_z z\right)\vec{v}\cdot d\vec{A} = \dot{Q}+\dot{W} \tag{3.3}$$

where t is time, ρ is the fluid density, V is the fluid volume within the control volume, ε is the control volume porosity (for porous media flow modeling), v is an average fluid velocity, A is the control surface area, S is the fluid mass source term, p is the static (or thermodynamic) pressure, $\dot{m}$ is the mass flow rate, E_f is the mechanical energy loss due to viscous forces, u is the fluid specific internal energy, g_z is the vertical component of the acceleration vector ($\vec{g}$), z is the elevation above some reference datum, $\dot{Q}$ is the rate of heat added to the control volume (by conduction, convection, and direct volumetric heating), and $\dot{W}$ is the rate of work done on the control volume. The *cv* and *cs* limits under the integral signs indicate integration over the control volume and control surface, respectively. An arrow over a variable indicates a vector quantity. The mass flow rate in Equation (3.2) is defined as

$$\dot{m} = \rho v \varepsilon A \tag{3.4}$$

with the variables as defined previously. The control surface area A is usually referred to as a superficial flow area; it is the area available for flow if there were no porous media within the control volume. The quantity εA is referred to as the pore area, εv is referred to as the superficial fluid velocity, and v is referred to as the pore fluid velocity. Because mass flow rate is a convenient engineering quantity, it is used as a dependent variable in SAFSIM instead of velocity.

In the next three subsections, all terms of the conservation equations are discussed, including details of how the integrations are performed. Also, all assumptions related to the final form of the equations are presented.

3.1 Conservation of Mass

The conservation of mass equation, expressed qualitatively, requires that the rate of change of mass in the control volume plus the net mass efflux through the control surfaces equals the rate of mass creation or destruction (i.e., the source term). Thus the creation or destruction of mass is allowed. If an average density for the control volume is defined as

$$\bar{\rho} = \frac{\int_{CV} \rho dV}{V} \tag{3.5}$$

then the first term in Equation (3.1) can be written as $V\frac{\partial \rho}{\partial t}$ where the overbar is omitted from the density variable for brevity. Thus ρ is now a representative average density for the control volume. For a 1-D fixed control volume with two ports (one inlet and one outlet), the second term of Equation (3.1), after performing the integration, becomes $(\rho v \varepsilon A)_2 - (\rho v \varepsilon A)_1$. The 1 and 2 subscripts indicate the inlet and outlet surfaces (ports) of the control volume, respectively. This result is obtained recognizing that the dot product of the velocity and differential area vectors is zero everywhere except where the velocity vector is normal to the differential area vector; thus nonzero terms occur only at the inlet and outlet ports. It should also be recognized that the 1-D assumption implies uniform properties at the inlet and outlet ports. Equation (3.1) can now be written as

$$V\frac{\partial\rho}{\partial t}+\dot{m}_2-\dot{m}_1=S \tag{3.6}$$

This is the 1-D conservation of mass equation used in SAFSIM. The major simplifying assumption is that of one dimensionality. It should be noted that this equation is never solved directly for the density field. The density is determined using an equation of state based on the calculated pressure and temperature fields. The equation of state and the conservation equations are solved iteratively to ensure simultaneous solution of all dependent variables.

Two new variables are introduced at this time:

$$\chi_1=\frac{\dot{m}_1}{\dot{m}} \text{ and } \chi_2=\frac{\dot{m}_2}{\dot{m}} \tag{3.7}$$

These variables are called the mass flow rate fractions. Also, the following relationship between the mass flow rate of the control volume and the mass flow rates at the inlet and outlet ports is presented here as

$$w_m\dot{m}_1+(1-w_m)\dot{m}_2=\dot{m} \tag{3.8}$$

where w_m is a mass flow rate weighting factor with a value between 0 and 1; a value of 1/2 is typical. Combining Equations (3.6), (3.7), and (3.8) results in the following expressions for the mass flow rate fractions:

$$\chi_2=1+\frac{w_m}{\dot{m}}\left(S-V\frac{\partial\rho}{\partial t}\right) \text{ and } \chi_1=1-\frac{(1-w_m)}{\dot{m}}\left(S-V\frac{\partial\rho}{\partial t}\right) \tag{3.9}$$

Use of these expressions is convenient in the formulation of the mechanical energy equation. They are used in an iterative solution procedure for the fluid mechanics equations that is discussed in detail later.

The differential equation for conservation of mass corresponding to Equation (3.6) is presented here for reference:

$$\frac{\partial\rho}{\partial t}+\frac{\partial(\rho v)}{\partial x}=\frac{S}{V} \tag{3.10}$$

This is the equation typically used in finite difference computer programs.

3.2 Conservation of Momentum

The integral balance equation for the conservation of momentum (which is a statement of Newton's second law) is given by

$$\frac{\partial}{\partial t}\int_{CV}\rho v dV + \int_{CS}\varepsilon\vec{v}\rho\vec{v}\cdot d\vec{A} = \int_{CV}\rho\cdot\vec{g}dV - \int_{CV}\nabla p dV - \varepsilon\rho A E_f \tag{3.11}$$

The mechanical energy equation is derived by taking the dot product of the velocity vector with the momentum equation. Thus the mechanical energy equation governs kinetic energy and is not independent of the momentum equation. (The phrases momentum equation and mechanical energy equation are used interchangeably throughout this document.) The momentum equation is expressed in terms of kinetic energy for mathematical convenience and to facilitate formulation of the finite element equations to follow in a later section. The momentum equation in terms of kinetic energy [Equation (3.2)], i.e., the mechanical energy equation, is repeated here for convenience:

$$\underbrace{\frac{\partial}{\partial t}\int_{CV}\tfrac{1}{2}\rho v^2 dV}_{\text{term K1}} + \underbrace{\int_{CS}\tfrac{1}{2}\varepsilon\rho v^2\vec{v}\cdot d\vec{A}}_{\text{term K2}} = \underbrace{\int_{CV}\rho\vec{v}\cdot\vec{g}dV}_{\text{term K3}} - \underbrace{\int_{CV}\vec{v}\cdot\nabla p dV}_{\text{term K4}} - \underbrace{\dot{m}E_f}_{\text{term K5}} \tag{3.2}$$

Expressed qualitatively, the mechanical energy equation requires that the rate of change of kinetic energy in the control volume plus the net efflux of kinetic energy through the control surfaces equals the sum of the rate of work on the control volume due to surface and body forces (gravity body force, hydrostatic surface force, and viscous shear surface force).

Term K1 is the time rate of change of kinetic energy in the control volume. Analogous to the average density defined by Equation (3.5) and used in the time-derivative term of the mass conservation equation, an average kinetic energy for the control volume can be defined as

$$\tfrac{1}{2}\overline{\rho v^2} = \frac{\int_{CV}\tfrac{1}{2}\rho v^2 dV}{V} \tag{3.12}$$

Now,

$$\text{term K1} = \tfrac{1}{2}V\frac{\partial\left(\overline{\rho v^2}\right)}{\partial t} \approx \tfrac{1}{2}V\left[\rho\frac{\partial v^2}{\partial t} + v^2\frac{\partial\rho}{\partial t}\right] \tag{3.13}$$

The last operation was carried out assuming that the average of the product of ρ and v^2 equals the product of the averages ($\overline{\rho v^2} = \overline{\rho}\,\overline{v^2}$). This common assumption is reasonable if the quantities do not vary greatly over the control volume. In regions where the gradients are steep, smaller control volumes are warranted. As with any discrete volume formulation, volume-size parametric studies are required to determine appropriate volume sizes for any given problem. The overbars are omitted from the terms on the right hand side of Equation (3.13) for brevity.

Taking the derivative of the mass flow rate [Equation (3.4)] with respect to time, assuming the pore area is constant, yields

$$\frac{\partial \dot{m}}{\partial t} = \varepsilon A\left[\rho\frac{\partial v}{\partial t} + v\frac{\partial \rho}{\partial t}\right] \tag{3.14}$$

Also, the fluid volume is defined as

$$V = \varepsilon AL \tag{3.15}$$

where L is the flow length of the control volume. (Note that a variable without a subscript refers to an average or representative value for the control volume, whereas a variable with a 1 or 2 subscript refers to the value corresponding to either the inlet or outlet port of the volume.) Combining the expression for volume with Equations (3.13) and (3.14), along with some simple mathematical manipulations, results in the following equation for term K1 of the mechanical energy equation:

$$\text{term K1} = \frac{\dot{m}L}{\rho\varepsilon A}\left[\frac{\partial \dot{m}}{\partial t} - \frac{\dot{m}}{2\rho}\frac{\partial \rho}{\partial t}\right] \tag{3.16}$$

This term represents the unsteady kinetic energy of the fluid in the control volume and includes mass flow rate and density time-derivative terms.

Next, an expression for term K2 of the mechanical energy equation is determined. First, the integral over the control surface is broken into one integral over the inlet surface area and one integral over the outlet surface area (recall that the dot product of the velocity and the differential area vectors is zero everywhere except where the velocity vector is normal to the differential area vector). Thus

$$\text{term K2} = \int_{A_2}\tfrac{1}{2}\varepsilon\rho v^3 dA - \int_{A_1}\tfrac{1}{2}\varepsilon\rho v^3 dA \tag{3.17}$$

As mentioned previously, the 1-D assumption implies uniform properties at the inlet and outlet surfaces. However, for viscous flow, the velocity profile can not be uniform because of the shear stress between the fluid and the wall of the control volume. Thus the fluid velocity is a function of the distance from the wall. The integrals of term K2 can be simplified, however, by defining a kinetic energy correction factor, α. This factor is defined by the following equation:

$$\alpha = \frac{\int_A \varepsilon \rho v^3 dA}{\dot{m}\bar{v}^2} \tag{3.18}$$

where $\bar{v}$ is an average velocity over the area. Omitting the overbar from the velocity variable, term K2 is written simply as

$$\text{term K2} = \tfrac{1}{2}\alpha_2\left(\dot{m}v^2\right)_2 - \tfrac{1}{2}\alpha_1\left(\dot{m}v^2\right)_1 \tag{3.19}$$

Again, the 1 and 2 subscripts refer to the inlet and outlet ports, respectively.

The kinetic energy correction factor corrects the kinetic energy calculated using the average velocity, v. This engineering approach allows the kinetic energy to be written in terms of an average velocity and a correction factor. However, values for α must now be found according to Equation (3.18). For laminar flow in a circular pipe, the velocity profile can be assumed to be a parabolic function of the radius and independent of the Reynolds number (*Re*). Equation (3.18) can then be evaluated, resulting in α = 2. For fully-developed turbulent flow, as the Reynolds number (based on the diameter as the characteristic length) increases, the profile becomes steeper near the walls and flatter (more blunt) away from the walls (see Figure 2). It turns out that the velocity profile is a weak function of the Reynolds number as indicated in Table 2. Because α varies very little for turbulent flow, a single value of 1.05 is used in SAFSIM for turbulent flow while a value of 2 is used for laminar flow. Also, linear interpolation based on the Reynolds number is used in the transition region. [In SAFSIM, the user specifies the Reynolds number above which the flow is turbulent (*Re*$_u$). The Reynolds number below which the flow is laminar (*Re*$_l$) is then determined as 0.6·*Re*$_u$.] For porous media flow, α = 1 for all *Re*.

Table 2. Kinetic Energy Correction Factor

Re	α
< 2,300	2.0000
4,000	1.0770
23,000	1.0648
$1.1 \cdot 10^5$	1.0580
$1.1 \cdot 10^6$	1.0387
$3.3 \cdot 10^6$	1.0306

Because of the small variations of α with *Re*, the additional assumption that $\alpha_1 = \alpha_2 = \alpha$ is made. Thus α is based on the Reynolds number for the control volume and is assumed to be uniform within the volume. The equation for α is

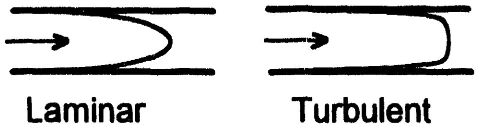


Figure 2. Example Radial Velocity Profiles in a Pipe

valid strictly only for incompressible flow; however, using this equation for compressible flow introduces very little error if the control volumes are sufficiently small. Thus Equation (3.19) simplifies to

$$\text{term K2} = \tfrac{1}{2}\alpha\left[\dot{m}_2 v_2^2 - \dot{m}_1 v_1^2\right] \tag{3.20}$$

Now the expression for the mass flow rate is substituted into Equation (3.20) along with the mass flow rate fractions [Equation (3.7)] and the assumption that $\varepsilon_1 = \varepsilon_2 = \varepsilon$. Term K2 of the mechanical energy equation now becomes

$$\text{term K2} = \dot{m}\frac{\alpha\dot{m}^2}{2\varepsilon^2}\left[\frac{\chi_2^3}{\rho_2^2 A_2^2} - \frac{\chi_1^3}{\rho_1^2 A_1^2}\right] \tag{3.21}$$

This term represents the mechanical energy due to changes in velocity and is associated with what is usually called the dynamic pressure or velocity pressure. The introduction of the mass flow rate fraction variables allows the control volume mass flow rate to be factored outside the brackets. These variables are defined in Subsection 3.1 and are discussed in more detail later.

Term K3 represents the mechanical energy associated with the gravity body force (potential energy). Only an axial component of acceleration is included. To simplify this term, first the differential volume is written in terms of the control volume differential length, *dx*. Thus

$$dV = \varepsilon A dx \tag{3.22}$$

Now, term K3 can be written as

$$\text{term K3} = \int_{CL} \rho\vec{v}\cdot\vec{g}\varepsilon A dx = \int_{z_1}^{z_2} -\rho v g_z \varepsilon A dz \tag{3.23}$$

where CL indicates integration over the length of the control volume, and z_1 and z_2 are the relative elevations of the control volume inlet and outlet ports, respectively. Applying the expressions for mass flow rate and the mass flow rate fractions, and assuming g_z doesn't vary with position, yields

$$\text{term K3} = -g_z(\dot{m}_2 z_2 - \dot{m}_1 z_1) = -g_z\dot{m}(\chi_2 z_2 - \chi_1 z_1) \qquad (3.24)$$

For liquids, the values of the mass flow rate fractions differ very little from unity. For gases, the potential energy term (K3) is usually very small compared to the other terms in the mechanical energy equation; thus assuming that the mass flow rate fractions both equal unity does not introduce appreciable error for most engineering flow systems of interest. The equation for term K3 can then be simplified to

$$\text{term K3} = -\dot{m}g_z(z_2 - z_1) = -\dot{m}g_z\Delta z \qquad (3.25)$$

where Δz is the change in elevation from outlet to inlet. Thus only Δz has to be specified for each control volume.

Making use of the differential volume from Equation (3.22) allows term K4 of the mechanical energy equation to be written as

$$\text{term K4} = \int_{CL} v\left(\frac{\partial p}{\partial x}\right)\varepsilon A dx = \int_{p_1}^{p_2} v\varepsilon A dp \qquad (3.26)$$

where p is the static pressure. The quantity $v\varepsilon A$ is the volumetric flow rate, $\dot{w}$. Now analogous to the mass flow rate fractions, the volumetric flow rate fractions are defined as

$$\lambda_1 = \frac{\dot{w}_1}{\dot{w}} \text{ and } \lambda_2 = \frac{\dot{w}_2}{\dot{w}} \qquad (3.27)$$

The integral of Equation (3.26) can then be approximated as

$$\text{term K4} = \frac{\dot{m}}{\rho}[\lambda_2 p_2 - \lambda_1 p_1] \qquad (3.28)$$

Use of the volumetric flow rate fractions allows the volumetric flow rate for the control volume to be factored outside the parentheses, which is then expressed in terms of the mass flow rate divided by the density.

As an aside, if λ_1 and λ_2 are assumed to equal 1 (steady, incompressible flow), the solution for p is greatly simplified. For liquids, this is a reasonable assumption. For gases, the incompressible assumption for all terms of the mechanical energy equation is a good assumption if the Mach number is less than about 0.3. If the incompressible assumption is made only for term K4, reasonable solutions can be obtained for all subsonic flows. It is not a good assumption, however, for regions of very large gradients, such as those that exist in a shock wave. This also is discussed in more detail later.

Returning to Equation (3.28), this equation can be expressed in terms of the mass flow rate fractions as

$$\text{term K4} = \dot{m}\left[\frac{\chi_2}{\rho_2}p_2 - \frac{\chi_1}{\rho_1}p_1\right] \tag{3.29}$$

The final term of the mechanical energy equation involves the energy losses due to viscous dissipation. The governing physics for this dissipation is very involved; however, in a 1-D solution of the fluid mechanics equations the use of empirical correlations (based on experiment) replaces this physics. This common engineering approach is expressed in the following equation for the mechanical energy loss, E_f:

$$E_f = f\left[\frac{L}{D} + \left(\frac{L}{D}\right)_e\right]\frac{(\varepsilon v)^2}{2} + K\frac{(\hat{\varepsilon v})^2}{2} \tag{3.30}$$

where f is the Darcy friction factor, L is the flow length, D is the equivalent diameter of the control volume, $\left(\frac{L}{D}\right)_e$ is an equivalent length-to-diameter ratio to account for minor losses such as pipe bends, εv is the superficial fluid velocity, K is an added loss coefficient (sometimes referred to as a form loss coefficient or as a K factor) to account for minor losses such as expansions and contractions, and $\hat{\varepsilon v}$ is the maximum superficial fluid velocity occurring within the control volume. The added loss coefficient is flow direction dependent according to the following rules:

$$K = K_{1-2} \text{ if } \dot{m} \geq 0 \text{ and } K = K_{2-1} \text{ if } \dot{m} < 0 \tag{3.31}$$

where $K_{1\text{-}2}$ is the added loss coefficient for flow from port 1 to port 2 (defined as the positive flow direction), and $K_{2\text{-}1}$ is for flow from port 2 to port 1 (negative flow). This allows the analyst to account for the different loss coefficients associated with flow in different directions. Equation (3.30) is then written in terms of the mass flow rate and density, to be consistent with the other terms.

$$E_f = \frac{\dot{m}^2}{2\rho^2}\left\{\frac{f}{A^2}\left[\frac{L}{D}+\left(\frac{L}{D}\right)_e\right]+\frac{K}{A_{\min}^2}\right\} \tag{3.32}$$

where $A_{\min}$ is the minimum flow area of the control volume (either at port 1 or port 2). The use of the engineering factors (friction factor, added loss coefficient, and equivalent length-to-diameter ratio) in the mechanical energy loss term is common in 1-D fluid mechanics modeling. Term K5 is now given as

$$\text{term K5} = \dot{m}\frac{\dot{m}^2}{2\rho^2}\left\{\frac{f}{A^2}\left[\frac{L}{D}+\left(\frac{L}{D}\right)_e\right]+\frac{K}{A_{\min}^2}\right\} \tag{3.33}$$

Before proceeding to a discussion of the friction factor, *f*, a few comments about Equation (3.32) are in order.

- The loss term is expressed in terms of the superficial fluid velocity because it is consistent with the available friction factor correlations for flow in porous media. For flow in a pipe, the porosity equals 1 and Equation (3.32) remains valid.

- *D* is an equivalent diameter for the control volume. (The variable *D* for control volumes containing porous media is discussed later in this subsection.) For control volume geometries not too far from circular or for flow parallel to a bank of tubes or rods, the hydraulic diameter is appropriate and is defined as

$$D_h = \frac{4A}{P_w} \tag{3.34}$$

  where D_h is the hydraulic diameter, and P_w is a wetted perimeter; i.e., the perimeter of the control volume in contact with the fluid. For irregular geometries, the hydraulic diameter should be multiplied by a correction factor to determine the equivalent diameter. Reference 3 contains tables of the appropriate correction factors for concentric annulus, rectangular, and triangular geometries. For example, for flow in a concentric annulus with an inside-to-outside radius ratio of 0.8, the correction factor is 0.667. For a square flow passage, the correction factor is 1.125.

- A wealth of tables and correlations is available in the literature for determining values for *K* and $(L/D)_e$ for a large variety of flow geometries and configurations. For example, Reference 6 is a popular handbook dedicated entirely to such loss coefficients. These coefficients account for multidimensional flow patterns that cannot be resolved with a 1-D solution.

However, at the system level, the details of such flow patterns are of less concern than the overall pressure loss associated with them. If desired, SAFSIM calculates the added loss coefficients for gradual and abrupt area changes associated with contractions and expansions based on correlations from Reference 6. These correlations and their implementation in SAFSIM are provided in Appendix A.

- The added loss coefficients (K) found in the literature are usually based on the maximum velocity. For incompressible flow, the maximum velocity occurs at the minimum area; hence the use of $A_{\min}$ in Equation (3.32). For compressible flow, this may not be the case. The approach used in Equation (3.32) is still reasonable, however, considering that relatively large uncertainties are usually associated with the experimentally determined values. Also, most loss coefficients are only applicable to incompressible flow.

Like the loss coefficients, a wealth of data exists for determination of the friction factor; the friction factor data of Moody[7] being perhaps the most famous. An empirical fit of the Moody friction factor data is provided in Reference 8. This fit is used in SAFSIM for turbulent flow in rough pipes and is provided here:

$$f = 1.325\left[\ln\left(0.27027\frac{\epsilon}{D} + 5.74Re^{-0.9}\right)\right]^{-2} \qquad (3.35)$$

where ϵ is the wall roughness, (ϵ/D is referred to as the relative roughness), and Re is again the Reynolds number using the equivalent diameter as the characteristic length. This correlation is used for relative roughness values greater than or equal to $1.0 \cdot 10^{-6}$.

For turbulent flow in smooth pipes ($\epsilon/D < 1.0 \cdot 10^{-6}$), the correlation chosen for use in SAFSIM is that of Petukhov[9]:

$$f = \left(1.82\log Re - 1.64\right)^{-2} \qquad (3.36)$$

This correlation was selected because it is a relatively recent correlation that agrees well with the data from many experimenters and because a corresponding heat transfer coefficient is provided as a function of this friction factor.

For isothermal laminar flow, the friction factor can be derived by assuming a parabolic velocity profile. This profile is used in conjunction with the definition of the Darcy friction factor (which is proportional to the fluid shear stress at the wall for Newtonian fluids) to yield the following classic result:

$$f = \frac{64}{Re} \tag{3.37}$$

Corrections can be made to this equation for large temperature differences between the wall and the fluid. However, the extra computational effort is probably not justified for most engineering analyses because of the small dependence. Also, the friction factor in turbulent flow is not a function, or is only a very weak function, of the temperature difference between the wall and fluid.

For the transition flow regime ($Re_l \le Re \le Re_u$), linear interpolation between the laminar and turbulent friction factors is determined as

$$f = w_f f_l + \left(1 - w_f\right) f_t \tag{3.38}$$

where f_l and f_t are the friction factors corresponding to laminar [Equation (3.37)] and turbulent flow [Equation (3.35) or (3.36)], respectively; and the friction weighting factor, w_f, is given as

$$w_f = \frac{Re_u - Re}{Re_u - Re_l} \tag{3.39}$$

This is the same interpolation scheme used for the kinetic energy correction factor discussed earlier in this subsection. Use of this interpolation scheme helps to smooth the transition between flow regimes and facilitates convergence of the numerical solution algorithm.

The friction factor for flow in porous media has the following form:

$$f = \left(\frac{1-\varepsilon}{\varepsilon^3}\right)\left[c_2\left(\frac{1-\varepsilon}{Re}\right)^{c_4} + c_3\left(\frac{1-\varepsilon}{Re}\right)^{c_5} + c_1\right] \tag{3.40}$$

where the Reynolds number is based on the superficial flow area and the characteristic length is an effective particle diameter. Thus

$$Re = \frac{\dot{m}D}{A\mu} \tag{3.41}$$

where the equivalent diameter D is now an effective particle diameter, and μ is the fluid dynamic viscosity. [With A defined as the superficial flow area, Equation (3.41) is appropriate for flow in porous media and in pipes.] For non-spherical particles of a porous media, D can be defined as

$$D = 6\frac{v_p}{s_p} \tag{3.42}$$

where $\frac{v_p}{s_p}$ is the volume-to-surface area ratio of the particle. For particles of different sizes, a volume-weighted average of the volume-to-surface area ratio can be used as long as the size variation is not too great. The constants appearing in Equation (3.40), c_1 through c_5, are provided in Table 3 for several popular porous media correlations.

Table 3. Porous Media Friction Factor Correlation Constants

		Constants				
Correlation	**Comments**	c_1	c_2	c_3	c_4	c_5
Ergun[10]	original version	3.5	300	0	1	1
Modified Ergun[11]	smooth surfaces	3.6	360	0	1	1
Modified Ergun[11]	rough surfaces	8	360	0	1	1
Achenbach[12]	-	1.75	320	20	1	0.4

SAFSIM allows the analyst to select any of these correlations for each control volume. Or, the analyst can elect to define different correlations by specifying the five *c* constants of Equation (3.40) for each additional correlation.

It should be noted that all of the friction factor correlations are based on steady-state experimental data. Nevertheless, the correlations are used for transient calculations due to lack of sufficient data to support general transient correlations. This approach is common in fluid mechanics modeling and in most cases is satisfactory.

The last step in the formulation of the mechanical energy equation is to combine the five terms just developed [Equations (3.16), (3.21), (3.25), (3.29), and (3.33)]. This produces, after dividing out a mass flow rate from each term, the following equation:

$$\begin{aligned} &\frac{L}{\rho\varepsilon A}\left[\frac{\partial\dot{m}}{\partial t} - \frac{\dot{m}}{2\rho}\frac{\partial\rho}{\partial t}\right] + \frac{\alpha\dot{m}^2}{2\varepsilon^2}\left[\frac{\chi_2^3}{\rho_2^2A_2^2} - \frac{\chi_1^3}{\rho_1^2A_1^2}\right] = \\ &-g_z\Delta z - \left[\frac{\chi_2}{\rho_2}p_2 - \frac{\chi_1}{\rho_1}p_1\right] - \frac{\dot{m}^2}{2\rho^2}\left\{\frac{f}{A^2}\left[\frac{L}{D} + \left(\frac{L}{D}\right)_e\right] + \frac{K}{A_{\min}^2}\right\} \end{aligned} \tag{3.43}$$

The pseudo pressures $\tilde{p}_1$ and $\tilde{p}_2$ are defined now to simplify this equation:

$$\tilde{p}_1 = \frac{\chi_1}{\rho_1} p_1 \quad \text{and} \quad \tilde{p}_2 = \frac{\chi_2}{\rho_2} p_2 \tag{3.44}$$

Equation (3.43) is then rearranged as follows:

$$\begin{aligned}&[\tilde{p}_2 - \tilde{p}_1] + g_z \Delta z + \frac{L}{\rho \varepsilon A}\left[\frac{\partial \dot{m}}{\partial t} - \frac{\dot{m}}{2\rho}\frac{\partial \rho}{\partial t}\right] = \\ &\frac{\dot{m}^2}{2}\left[\frac{\alpha}{\varepsilon^2}\left(\frac{\chi_1^3}{\rho_1^2 A_1^2} - \frac{\chi_2^3}{\rho_2^2 A_2^2}\right) - \frac{1}{\rho^2}\left\{\frac{f}{A^2}\left[\frac{L}{D} + \left(\frac{L}{D}\right)_e\right] + \frac{K}{A_{\min}^2}\right\}\right]\end{aligned} \tag{3.45}$$

Now, let

$$\Re - \frac{\dot{m}}{2}\left[\frac{\alpha}{\varepsilon^2}\left(\frac{\chi_1^3}{\rho_1^2 A_1^2} - \frac{\chi_2^3}{\rho_2^2 A_2^2}\right) - \frac{1}{\rho^2}\left\{\frac{f}{A^2}\left[\frac{L}{D} + \left(\frac{L}{D}\right)_e\right] + \frac{K}{A_{\min}^2}\right\}\right] \tag{3.46}$$

where $\Re$ is an effective flow resistance that accounts for both velocity pressure losses (or gains) and viscous losses. Also, let the reciprocal of $\Re$ equal F, where F is called the fluidity. Equation (3.45) now becomes:

$$F[\tilde{p}_2 - \tilde{p}_1] = \dot{m} - F g_z \Delta z - \frac{FL}{\rho \varepsilon A}\left[\frac{\partial \dot{m}}{\partial t} - \frac{\dot{m}}{2\rho}\frac{\partial \rho}{\partial t}\right] \tag{3.47}$$

This is the mechanical energy equation derived from the control volume integral balance equation [Equation (3.2)]. Note that the use of the mass flow rate as a dependent variable in the mechanical energy equation results in two time-derivative terms appearing in Equation (3.47). Namely, the mass flow rate time derivative and the density time derivative. As a note of interest, for steady, incompressible, frictionless flow, this equation reduces to the well-known Bernoulli equation. The details of how Equation (3.47) is combined with the conservation of mass equation [Equation (3.1)] to find the pressure and mass flow rate fields is described in a later section.

3.3 Conservation of Energy

The integral balance equation for the conservation of energy is provided by Equation (3.3). Qualitatively, it states that the rate of change of energy in the control volume plus the net energy efflux through the control surfaces equals the sum of the energy added to and the rate of work done on the fluid in the control volume. This equation is repeated here for convenience.

$$\underbrace{\frac{\partial}{\partial t}\int_{CV}\rho\left(u+\tfrac{1}{2}v^2+g_z z\right)dV}_{\text{term E1}}+\underbrace{\int_{CS}\varepsilon\rho\left(u+\tfrac{1}{2}v^2+g_z z\right)\vec{v}\cdot d\vec{A}}_{\text{term E2}}=\underbrace{\dot{Q}}_{\text{term E3}}+\underbrace{\dot{W}}_{\text{term E4}} \qquad (3.3)$$

The first step in the simplification of this equation is to express term E4 as a sum of three separate work terms. Thus

$$\dot{W}=\dot{W}_{shaft}+\dot{W}_{vis}+\dot{W}_p \qquad (3.48)$$

where the subscripts *shaft*, *vis*, and *p* refer to shaft, viscous stress, and pressure, respectively. (The work of gravitational forces is already included in term E2.) The shaft work is of course the work performed on the fluid by the shaft of a mechanical device such as a pump or a turbine. The viscous shear work is expressed by the following equation:

$$\dot{W}_{vis}=\int_{CS}(\vec{\tau}\cdot\vec{v})d\vec{A} \qquad (3.49)$$

where $\vec{\tau}$ is the shear stress vector acting on the differential surface area. It is assumed that at a solid surface, the integral of Equation (3.49) is zero because the velocity is zero at a wall due to an assumed no-slip boundary condition. At the inlet and outlet surfaces, only the normal stress component is nonzero because the flow is assumed to be normal to these surfaces (i.e., 1-D flow). These viscous normal stresses are assumed to be negligible. This is a reasonable assumption except for the interior of a shock wave. SAFSIM is not intended for the study of a shock wave interior and thus the only nonzero contribution of the work term (other than any specified shaft work) is due to pressure forces. The pressure work term is given as

$$\dot{W}_p=-\int_{CS}p\,\vec{v}\cdot d\vec{A} \qquad (3.50)$$

where p is the normal component of stress and is referred to as the static pressure. For convenience, the pressure work term is added to term E2 of Equation (3.3), which is now written as

$$\underbrace{\frac{\partial}{\partial t}\int_{CV}\rho\left(u+\tfrac{1}{2}v^2+g_z z\right)dV}_{\text{term E1}}+\underbrace{\int_{CS}\varepsilon\rho\left(u+\tfrac{1}{2}v^2+g_z z+\frac{p}{\rho}\right)\vec{v}\cdot d\vec{A}}_{\text{term E2}}=\underbrace{\dot{Q}}_{\text{term E3}}+\underbrace{\dot{W}_{shaft}}_{\text{term E4}} \qquad (3.51)$$

Proceeding with the development of the conservation of energy equation, term E1 can be written as

$$\text{term E1} = V\frac{\partial(\rho u)}{\partial t} + V\frac{\partial\left(\frac{1}{2}\rho v^2\right)}{\partial t} + V\frac{\partial(\rho g_z z)}{\partial t} \tag{3.52}$$

where u is the fluid specific internal energy. This equation is derived using the volume-averaged kinetic energy defined in Equation (3.12) along with the following two new analogous definitions:

$$\overline{\rho u} = \frac{\int_{CV} \rho u dV}{V} \tag{3.53}$$

and

$$\overline{\rho g_z z} = \frac{\int_{CV} \rho g_z z dV}{V} \tag{3.54}$$

As done for the conservation of mass and momentum equations, the overbars indicating an averaged property are omitted for brevity. The product of averages is again assumed to equal the average of the products.

Further expansion of term E1 yields:

$$\text{term E1} = \rho V\frac{\partial u}{\partial t} + \rho v V\frac{\partial v}{\partial t} + \left(u + \tfrac{1}{2}v^2 + g_z z\right)V\frac{\partial \rho}{\partial t} \tag{3.55}$$

For most practical engineering problems, the specific potential energy ($g_z z$) is much less than the specific internal energy (u) and can be neglected.

From thermodynamics, the differential specific internal energy can be expressed as

$$du = \left(\frac{\partial u}{\partial T}\right)_{\upsilon} dT + \left(\frac{\partial u}{\partial \upsilon}\right)_T d\upsilon \tag{3.56}$$

where T is the fluid temperature and υ is the specific volume (not to be confused with the velocity, v). The υ and T subscripts indicate that the corresponding terms in parentheses are evaluated at constant volume and constant temperature, respectively. The term $\left(\frac{\partial u}{\partial T}\right)_v$ is given the special symbol, c_v; this is the constant-volume specific heat. For an ideal gas, the second term to the right of the equal sign is zero and for most liquids, the second term is very small compared to the first term. Therefore, for engineering purposes, Equation (3.56) can be approximated very well by

$$du = c_v dT \tag{3.57}$$

When integrated, this equation results in

$$u = \bar{c}_v T + \text{constant} \tag{3.58}$$

where the constant is assumed to equal zero, consistent with the common practice of assigning $u = 0$ at an absolute temperature of zero. Also, $\bar{c}_v$ is the constant-volume specific heat averaged over the applicable temperature range. It is common to determine $\bar{c}_v$ by evaluating the specific heat at the arithmetic mean of the associated lower and upper fluid temperature limits of integration for each control volume. As has been the practice throughout this document, the overbar average symbol is omitted from here on for brevity. Now, term E1 becomes

$$\text{term E1} = \rho V c_v \frac{\partial T}{\partial t} + \rho v V \frac{\partial v}{\partial t} + \left(c_v T + \tfrac{1}{2} v^2\right) V \frac{\partial \rho}{\partial t} \tag{3.59}$$

Combining this equation with Equation (3.14) for the mass flow rate time derivative and Equation (3.15) for volume results in

$$\text{term E1} = \rho V c_v \frac{\partial T}{\partial t} + c_v T V \frac{\partial \rho}{\partial t} + \frac{\dot{m} L}{\rho \varepsilon A} \frac{\partial \dot{m}}{\partial t} - \frac{\dot{m}^2 L}{2\rho^2 \varepsilon A} \frac{\partial \rho}{\partial t} \tag{3.60}$$

The reason for expressing the last two subterms as they are will become apparent when terms E1 through E4 are combined later. Term E1 is the time rate of change of energy in the control volume and contains time derivative terms for the three dependent variables ρ, $\dot{m}$, and T.

Term E2 of Equation (3.51) is the net efflux of energy through the control surfaces. Again, from thermodynamics

$$h = u + \frac{p}{\rho} \tag{3.61}$$

where h is the specific enthalpy, and

$$dh = \left(\frac{\partial h}{\partial T}\right)_p dT + \left(\frac{\partial h}{\partial p}\right)_T dp \tag{3.62}$$

where again, the p and T subscripts indicate that the corresponding terms in parentheses are evaluated at constant pressure and constant temperature, respectively. The term $\left(\frac{\partial h}{\partial T}\right)_p$ is given the special symbol, c_p; this is the constant-pressure specific heat. For an ideal gas, the second term to the right of the equal sign is zero and for most liquids, the second term is very small compared to the first term. Therefore, for engineering purposes, Equation (3.62) can be approximated very well by

$$dh = c_p dT \tag{3.63}$$

It is instructive to expand Equation (3.62) without neglecting the second term. Thus

$$dh = c_p dT + (1 - \beta_e T)\frac{dp}{\rho} \tag{3.64}$$

where β_e is the coefficient of thermal expansion (also called the volume expansivity) and is defined as

$$\beta_e = \frac{1}{v}\left(\frac{\partial v}{\partial T}\right)_p \tag{3.65}$$

For an ideal gas, β_e equals $1/T$ and the second term to the right of the equal sign in Equation (3.64) vanishes. For most liquids, although β_e is close to zero, dp/ρ is usually small compared to $c_p dT$. It is important to recognize that although Equation (3.63) is valid for a large class of fluids and flow situations, there are cases for which it is inadequate. Equations (3.64) and (3.65) provide information helpful in determining when Equation (3.63) is reasonable. Similar results can be obtained for the specific internal energy with the aid of Equation (3.61).

Continuing with Equation (3.63), integration of this equation results in

$$h = \bar{c}_p T + \text{constant} \tag{3.66}$$

where the constant is assumed to equal zero, consistent with assigning $u = 0$ at an absolute temperature of zero in Equation (3.58). The overbar average symbol is again omitted from c_p from here on. Now, term E2 becomes

$$\text{term E2} = c_p\left(\dot{m}_2 T_2 - \dot{m}_1 T_1\right) + \tfrac{1}{2}\alpha\left(\dot{m}_2 v_2^2 - \dot{m}_1 v_1^2\right) + g_z\left(\dot{m}_2 z_2 - \dot{m}_1 z_1\right) \qquad (3.67)$$

where the kinetic energy contribution is derived just like term K2 of the mechanical energy equation, and enthalpy is assumed to be uniform across the control surfaces, consistent with the 1-D assumption. Using the definitions of the mass flow rate fractions from Equation (3.7) and implementing the same assumptions regarding the potential energy contribution as imposed on term K3 of the mechanical energy equation results in

$$\text{term E2} = \dot{m}\left\{c_p\left(\chi_2 T_2 - \chi_1 T_1\right) + \frac{\alpha \dot{m}^2}{2\varepsilon^2}\left[\frac{\chi_2^{\,3}}{\rho_2^2 A_2^2} - \frac{\chi_1^{\,3}}{\rho_1^2 A_1^2}\right] + g_z \Delta z\right\} \qquad (3.68)$$

All of the details concerning derivation of this equation from Equation (3.67) are covered in Subsection 3.2 and therefore are not repeated here. Term E2 represents the net efflux of energy through the control surfaces. Terms E3 and E4 are just contributions to the power added to the fluid and therefore are combined into a single variable q, for convenience. Thus

$$q = \dot{Q} + \dot{W}_{shaft} \qquad (3.69)$$

Equations (3.60), (3.68), and (3.69) are now added together, which after some rearrangement yields the control volume conservation of energy equation:

$$\begin{gathered} \rho V c_v \frac{\partial T}{\partial t} + c_v T V \frac{\partial \rho}{\partial t} + \dot{m} c_p\left(\chi_2 T_2 - \chi_1 T_1\right) - q = \\ -\dot{m}\left\{\frac{L}{\rho \varepsilon A}\left[\frac{\partial \dot{m}}{\partial t} - \frac{\dot{m}}{2\rho}\frac{\partial \rho}{\partial t}\right] + \frac{\alpha \dot{m}^2}{2\varepsilon^2}\left[\frac{\chi_2^3}{\rho_2^2 A_2^2} - \frac{\chi_1^3}{\rho_1^2 A_1^2}\right] + g_z \Delta z\right\} \end{gathered} \qquad (3.70)$$

Equation (3.43) is now rearranged to produce the following convenient result:

$$\begin{gathered} -\dot{m}\left\{\frac{L}{\rho \varepsilon A}\left[\frac{\partial \dot{m}}{\partial t} - \frac{\dot{m}}{2\rho}\frac{\partial \rho}{\partial t}\right] + \frac{\alpha \dot{m}^2}{2\varepsilon^2}\left[\frac{\chi_2^3}{\rho_2^2 A_2^2} - \frac{\chi_1^3}{\rho_1^2 A_1^2}\right] + g_z \Delta z\right\} = \\ \dot{m}\left[\frac{\chi_2}{\rho_2} p_2 - \frac{\chi_1}{\rho_1} p_1\right] + \dot{m}\frac{\dot{m}^2}{2\rho^2}\left\{\frac{f}{A^2}\left[\frac{L}{D} + \left(\frac{L}{D}\right)_e\right] + \frac{K}{A_{\min}^2}\right\} \end{gathered} \qquad (3.71)$$

Comparison of Equation (3.70) to this version of the mechanical energy equation reveals that the expression on the right-hand side of Equation (3.70) is identical to the expression on the left-hand side of Equation (3.71). Now, based on the right-hand side of Equation (3.71), a new variable is introduced:

$$q_{flo} = \dot{m}\left\{\left[\frac{\chi_2}{\rho_2}p_2 - \frac{\chi_1}{\rho_1}p_1\right] + \frac{\dot{m}^2}{2\rho^2}\left(\frac{f}{A^2}\left[\frac{L}{D} + \left(\frac{L}{D}\right)_e\right] + \frac{K}{A_{\min}^2}\right)\right\} \tag{3.72}$$

where q_{flo} is the mechanical power associated with the flowing fluid that is converted to thermal power. It represents the recoverable an unrecoverable losses from compressibility and viscous dissipation and accounts for both kinetic and potential energy. The control volume conservation of energy equation is now

$$\rho V c_v \frac{\partial T}{\partial t} + c_v T V \frac{\partial \rho}{\partial t} + \dot{m} c_p(\chi_2 T_2 - \chi_1 T_1) = q + q_{flo} \tag{3.73}$$

This equation is referred to as the thermal energy equation and its solution yields the fluid temperature response. Defining another new variable, q_ρ as

$$q_\rho = -c_v T V \frac{\partial \rho}{\partial t} \tag{3.74}$$

where q_ρ is an effective power associated with density variations with time. Also, the power q is split into two components: q_d, which is the power added directly to the fluid (for example, neutron and gamma ray heating or chemical reaction heating), and q_k, which is the power associated with conduction in the fluid across control volume boundaries. The conduction power can be expressed as

$$q_k = V \xi k_t \frac{\partial^2 T}{\partial x^2} \tag{3.75}$$

where ξ is the fluid thermal conductivity enhancement factor (to be defined in Subsection 3.5) and k_t is the fluid thermal conductivity which is assumed to be independent of position. This is the net efflux of heat conducted into the control volume based on Fourier's law of conduction applied to the fluid. (Heat added to the fluid by convection heat transfer is also included in this term, via the natural boundary conditions, and will appear when the finite element procedures are implemented in a later section.) The thermal energy equation can now be expressed succinctly as

$$\rho V c_v \frac{\partial T}{\partial t} + \dot{m} c_p(\chi_2 T_2 - \chi_1 T_1) = q_d + q_k + q_{flo} + q_\rho \tag{3.76}$$

Simultaneous solution of the mechanical and thermal energy equations results in solution of the total energy equation, accounting for internal, pressure, kinetic, and potential energies. Equation (3.76) is equivalent to the following differential energy equation, which is sometimes used in finite difference computer programs:

$$\rho c_v \frac{\partial T}{\partial t} + c_p \frac{\partial(\rho v T)}{\partial x} = q_d''' + q_{flo}''' + q_\rho''' + \xi k_t \frac{\partial^2 T}{\partial x^2} \tag{3.77}$$

where the triple prime superscripts indicate per volume (or volumetric) quantities. Equations (3.76) and (3.77) can be shown to be equivalent by integrating Equation (3.77) over the volume. Although the variables q_{flo} and q_ρ are small and can be neglected for incompressible flow, they are crucial in compressible flow when Mach numbers are greater than about 0.3.

3.4 Multiple Fluid Mass Continuity

Although SAFSIM is currently a single-phase fluid computer program, it contains a multicomponent capability. This single-phase, multicomponent fluid capability allows multiple components of a single-phase mixed fluid to be tracked throughout a fluid mechanics network. This capability has a broad range of engineering applications. For example, hydrogen and helium gases can be modeled within a network, accounting for mixing of the two gases on a per finite element basis.

The multiple components of a mixed fluid are assumed to be in thermodynamic equilibrium and homogeneously mixed within an element. The governing equation for each component of the fluid is therefore simply a mass conservation equation. For a fluid component, the conservation of mass equation [Equation (3.6)] can be written as

$$x^k V \frac{\partial \rho}{\partial t} + x_2^k \dot{m}_2 - x_1^k \dot{m}_1 = x^k S \tag{3.78}$$

where x^k is the mass fraction for component k of the multicomponent fluid; thus

$$x^k = \frac{m^k}{m} \tag{3.79}$$

where m is the fluid mass in the control volume. The 1 and 2 subscripts in Equation (3.78) again refer to the inlet and outlet ports and a variable without a subscript refers to an average value for the control volume. The mass fractions are defined for the ports and control volume such that

$$\sum_{k=1}^{N_c} x^k = 1 \tag{3.80}$$

where N_C is the number of components comprising the mixed fluid. Thus Equation (3.78) need be solved only N_c-1 times. Substituting the mass flow rate fraction definitions of Equation (3.7) allows Equation (3.78) to be written as

$$x^k V \frac{\partial \rho}{\partial t} + x_2^k \dot{m} \chi_2 - x_1^k \dot{m} \chi_1 = S^k \tag{3.81}$$

where S^k is the mass source term for component k, such that

$$\sum_{k=1}^{N_c} S^k = S \tag{3.82}$$

where S is the total mass source for the control volume.

3.5 Equation of State and Fluid Properties

To complete the fluid mechanics governing equations, it is necessary to specify the fluid properties appearing within. The relevant fluid properties are density (ρ), dynamic viscosity (μ), specific heats (c_v and c_p), and thermal conductivity (k_f). The equation that governs the fluid density is referred to as the equation of state.

Three options are available in SAFSIM for the specification of fluid properties: (1) ideal gas, (2) ideal liquid, and (3) user-specified. An ideal liquid is a fluid in which its density is a function of temperature only and the constant-volume and constant-pressure specific heats are equal. The user-specified equation of state option allows the user to implement any equation of state and property data as a function of fluid pressure and temperature. Subroutine interfaces are supplied in SAFSIM that simplify the linking of any user-specified data. Because numerical solution procedures are adverse to discontinuities, care must be taken to ensure that the user-specified data is a smooth function of temperature and pressure. This helps to prevent oscillations and possible non-convergence of SAFSIM's iterative numerical methods.

The equation of state for an ideal gas is given as

$$\rho = \frac{p}{RT} \tag{3.83}$$

where R is the gas constant. For a multicomponent fluid comprised of ideal gases, the same equation is used with an effective R calculated as

$$R = \sum_{k=1}^{N_c} x^k R^k \tag{3.84}$$

where R^k is the gas constant for component k. If one of the fluid components is specified by a user-specified equation of state, Equation (3.84) is employed with an effective R^k for that density component determined by

$$R^k = \frac{p}{\rho^k T} \tag{3.85}$$

The partial pressures for a control volume are useful variables when dealing with multiple gases and are calculated as

$$p^k = p x^k \frac{R^k}{R} \tag{3.86}$$

Further details concerning Equations (3.84) and (3.86) can be found in Reference 13 and in most thermodynamic texts.

For an ideal liquid, the density is specified as a second-order polynomial function of fluid temperature; thus

$$\rho = \Lambda_1 + \Lambda_2 T + \Lambda_3 T^2 \tag{3.87}$$

where Λ_1, Λ_2, and Λ_3 are user-supplied constants. For a multicomponent fluid comprised of liquids, the mixture density is based on a simple mass fraction weighted average given by the following equation:

$$\rho = \sum_{k=1}^{N_c} x^k \rho^k \tag{3.88}$$

Two options are available for specification of the dynamic viscosity for both gases and liquids: the power-law and the Sutherland-law equations. The power-law equation is expressed as

$$\mu = \mu_0 \left(\frac{T}{T_0} \right)^{\psi} \qquad (3.89)$$

and the Sutherland-law equation is expressed as

$$\mu = \mu_0 \left(\frac{T}{T_0} \right)^{\frac{3}{2}} \left[\frac{T_0 + \psi}{T + \psi} \right] \qquad (3.90)$$

where μ_0, T_0, and ψ are user-supplied constants. These constants are supplied for many fluids of interest in various handbooks and texts such as *Viscous Fluid Flow*.[14] SAFSIM input allows specification of different constants for different temperature ranges. This multitemperature-range property capability provides added flexibility in property modeling.

For multicomponent fluids comprised of gases, the Method of Wilke as modified by Herning and Zipperer (outlined in Reference 15) is employed as the viscosity mixing rule:

$$\mu = \sum_{k=1}^{N_c} \frac{x^k R^k \mu^k}{\sum_{l=1}^{N_c} x^l R^l \left(\frac{R^k}{R^l} \right)^{\frac{1}{2}}} \qquad (3.91)$$

This method has been found satisfactory for a variety of mixtures, but may be inadequate for gas components with vastly different viscosities. For multicomponent fluids comprised of liquids, simple mass fraction weighting is used:

$$\mu = \sum_{k=1}^{N_c} x^k \mu^k \qquad (3.92)$$

The fluid properties used in the conservation equations are based on the average control volume fluid temperature which is sometimes called the bulk temperature. In the evaluation of heat transfer coefficients for use in Newton's law of cooling, a viscosity based on the fluid temperature at the wall (exchange surface) is sometimes required. In such instances, the wall viscosity is determined according to the following approximate relations:

$$\mu_w = \mu\left(\frac{T_w}{T}\right)^{0.7} \tag{3.93}$$

and

$$\mu_w^{-0.2661} = \mu^{-0.2661} + \frac{(T_w - T)}{37.073} \tag{3.94}$$

where μ_w is the viscosity based on the fluid temperature at the wall given by T_w. The constants in Equation (3.94) are appropriate for temperature in Kelvins and viscosity in Pa·s. The user can select between these two equations; Equation (3.93) is appropriate for gases and Equation (3.94) is appropriate for liquids. Both of these equations can be found in Reference 15.

The constant-pressure specific heat for both gases and liquids is expressed as a third-order polynomial function of fluid temperature. Thus

$$c_p = c_{p0} + c_{p1}T + c_{p2}T^2 + c_{p3}T^3 \tag{3.95}$$

where c_{p0}, c_{p1}, c_{p2}, and c_{p3} are user-supplied constants. Again, different constants for different temperature ranges can be used if desired. For an ideal gas,

$$c_v = c_p - R \tag{3.96}$$

and for an ideal liquid, $c_v = c_p$. For a user-specified fluid, c_v must be supplied.

For a multicomponent fluid (gas or liquid), mass fraction weighting of both c_v and c_p is used to determine the mixed properties. Thus

$$c_p = \sum_{k=1}^{N_c} x^k c_p^k \quad \text{and} \quad c_v = \sum_{k=1}^{N_c} x^k c_v^k \tag{3.97}$$

The thermal conductivity is also expressed as a third-order polynomial function of fluid temperature. Thus

$$k_t = k_{t0} + k_{t1}T + k_{t2}T^2 + k_{t3}T^3 \tag{3.98}$$

where k_{t0}, k_{t1}, k_{t2}, and k_{t3} are user-supplied constants (for different temperature ranges if desired). For multicomponent gases, the same mixing model employed for viscosity is used as suggested in Reference 15; thus

$$k_t = \sum_{k=1}^{N_c} \frac{x^k R^k k_t^k}{\sum_{l=1}^{N_c} x^l R^l \left(\frac{R^k}{R^l}\right)^{\frac{1}{2}}} \tag{3.99}$$

For ideal liquids, simple mass fraction weighting is used; thus

$$k_t = \sum_{k=1}^{N_c} x^k k_t^k \tag{3.100}$$

The fluid thermal conductivity enhancement factor, ξ, [introduced in Equation (3.75)] accounts for the enhancement of fluid conductivity due to dispersion effects as the fluid flows over and through a porous media and is given by the following equation:[16]

$$\xi = \frac{k_{t\xi}}{k_t} = \phi_p Re Pr \tag{3.101}$$

where $k_{t\xi}$ is the enhanced fluid thermal conductivity, and ϕ_p is the dispersion conductivity coefficient for flow in porous media. According to Reference 16, ϕ_p is equal to 1/2. However, other values of ϕ_p have been proposed. Also, Equation (3.101) has no explicit dependence on porosity. Therefore, SAFSIM allows the analyst to supply a value of ϕ_p for each porous media finite element, with 1/2 as the default value. For nonporous media elements, ξ is equal to 1.

An additional fluid property that is determined in SAFSIM is the local speed of sound or sonic velocity. For an ideal gas, the speed of sound, a, is given as

$$a = \sqrt{\frac{c_p}{c_v} RT} = \sqrt{\gamma RT} \tag{3.102}$$

where γ is the ratio of specific heats. For an ideal liquid, the speed of sound is

$$a = \sqrt{\frac{E_s}{\rho}} \tag{3.103}$$

where E_s is the isentropic bulk modulus of elasticity. For a user-specified fluid, the user must supply an appropriate R value (as a function of T and p if desired) such that Equation (3.102) is satisfied.

4.0 Fluid Mechanics Finite Element Formulation

The governing equations for SAFSIM fluid mechanics are derived in Section 3.0. The next step is to express the governing equations in terms of algebraic expressions that can be programmed on a computer. Perhaps the most common approach to this step is the finite difference method in which the derivative terms are expressed as algebraic differences; the resulting equations are then solved for the entire problem domain which has been divided into a group of connected geometric mesh intervals.

SAFSIM makes use of the finite element method, which is rapidly gaining popularity in the fluid mechanics community. This method has achieved great success in the structural mechanics community, where the method was originally developed. In this method, the problem domain is again divided into a group of mesh intervals (referred to as finite elements) and the governing equations are solved in an approximate integral sense for each individual finite element with some level of continuity enforced at the finite element connections (referred to as nodes). Thus the finite element method can be considered as an integral method. A fundamental difference between structural mechanics and fluid mechanics finite element models is that for structural mechanics, the finite element represents a material region whereas for fluid mechanics, the finite element represents a spatial region. Thus a finite element model of fluid motion is specified in terms of velocity *at* a node in space rather than *of* a node in the material.

A third method for solution of the fluid mechanics equations is the finite volume method, which can be considered to be a subset of the finite element method. There is a wealth of available literature describing the finite difference, finite element, and finite volume methods. Excellent introductions to the finite element method can be found in References 17 and 18. For one-dimensional applications, there are not significant mathematical differences between the three methods; however, the finite element method provides a convenient and well-defined framework for addressing the tedious bookkeeping chores required of complex system modeling, especially for a general-purpose program.

The finite element method allows the modeling of the whole as an assemblage of discrete parts or finite elements. Thus it is necessary to derive equations that express the mathematical properties of the individual elements. There are four approaches for deriving the finite element equations: (1) the direct approach, (2) the variational approach, (3) the weighted residuals approach, and (4) the energy balance approach.

The direct approach, so-called because the finite element equations are written based on direct physical reasoning, is the simplest of the four methods and offers an intuitive "feel" for the finite element process. This approach is limited to relatively simple problems. The variational approach relies on the calculus of variations and requires development of what is referred to as a variational principle. This method is more mathematically rigorous than the others but is seldom used in developing fluid mechanics computer programs because of the need to find an appropriate variational principle.

The third approach for deriving finite element equations is probably the most common and relies on the use of approximating (or interpolation) functions and weighting functions. This method of weighted residuals begins with a general functional relationship for the dependent variable (approximating function) that approximately satisfies the given differential equation and boundary conditions. Substitution of this approximating function into the original equation results in an error called the residual. A weighting function is then applied to this residual, which is required to vanish in some average sense over the entire solution domain. When the weighting function and the approximating function are the same, the method is referred to as the Bubnov-Galerkin or simply the Galerkin method. When the two functions are different, the method is referred to as the Petrov-Galerkin method. This latter method is employed in SAFSIM for the thermal energy equation.

The fourth approach relies on the use of a global energy balance to derive the finite element equations. This general method usually provides more insight into a problem and has the advantage that the energy balance can be specialized or generalized as the situation demands. In a sense, this approach is an extension of the direct approach based on a global balance equation. The energy balance governing equations are integrated over the solution domain; interpolation functions for the dependent variables are then imposed to arrive at the finite element equations. The energy balance approach for formulation of the finite element equations is also similar to the finite volume method which directly discretizes the integral form of the conservation equations.

The finite element equations for fluid mechanics used in SAFSIM are developed using the direct, weighted residuals, and energy balance approaches. For the thermal energy equation, the finite element equations are derived using what is known as the Petrov-Galerkin method. For the mechanical energy equation, the finite element equations are derived using what can be considered a combination of the direct and energy balance approaches. Use of the energy balance approach is why the fluid mechanics governing equations of Section 3 are expressed in terms of a mechanical energy equation and a thermal energy equation. The finite element equations for the thermal energy equation are derived first because some of their features are employed in the derivation of the finite element equations for the mechanical energy equation. The derivations are intended to provide sufficient information to allow interpretation and understanding of SAFSIM's capabilities. They are not intended to provide a complete tutorial on finite element theory.

4.1 Thermal Energy Equation

The thermal energy governing equation is known as an advective-conduction equation[+] and is the subject of extensive research and some controversy. It contains the combined effects of heat transfer due to a moving fluid (advection) and heat transfer due to fluid conduction. Mathematically, an advective equation is characterized as a hyperbolic equation while a conductive equation is characterized as an elliptic equation. The dual nature of this equation poses numerical difficulties which are addressed by use of an upwind formulation of the finite element equations.

The first step in the derivation of the thermal energy finite element equations, using the method of weighted residuals, is to integrate the governing differential equation (Equation 3.77) over the solution domain, Ω. The solution domain is broken into discrete elements, each of which contains N_n nodes. These finite elements are thus analogous to the control volumes used in the derivation of the governing equations and Ω can be considered as the finite element volume. Thus

$$\int_{\Omega} W_i \left[\rho c_v \frac{\partial T}{\partial t} + c_p \frac{\partial(\rho v T)}{\partial x} - q_t''' - \xi k_t \frac{\partial^2 T}{\partial x^2} \right] d\Omega = 0 \tag{4.1}$$

where W_i is the weighting function for the element and q_t''' is just the sum of the three volumetric heating terms $(q_d''' + q_{flo}''' + q_\rho''')$.

Integration of the conduction term is carried out using Gauss's theorem (integration by parts in 1-D). This produces two terms: a term involving integration over the volume, and a second term involving integration over the element boundary surface, Γ. Thus

$$\underbrace{\int_{\Omega} W_i \left[\xi k_t \frac{\partial^2 T}{\partial x^2} \right] d\Omega}_{\text{conduction term}} = \underbrace{\int_{\Gamma} W_i \left(\xi k_t \frac{\partial T}{\partial x} \right) d\Gamma}_{\text{surface integral}} - \underbrace{\int_{\Omega} \xi k_t \frac{\partial T}{\partial x} \frac{\partial W_i}{\partial x} d\Omega}_{\text{volume integral}} \tag{4.2}$$

The surface integral allows introduction of the following convective heat transfer boundary conditions:

[+] This equation is also referred to as the convective-diffusion equation. In this document, the term "advective" is used in place of "convective" and the term "conduction" (which is a diffusive process) is used in place of "diffusion."

$$-\xi k_t \frac{\partial T}{\partial x}\bigg|_{\Gamma} = \sum_{l=1}^{N_{ex}} \left[h_c(T_w - T)\right]_l \qquad (4.3)$$

where N_{ex} is the number of convective exchange surfaces (as requested by the analyst), h_c is the convective heat transfer coefficient for the exchange surface, and T_w is the wall temperature of the exchange surface (this is determined by solution of the structure heat transfer conduction equation to be developed in a later section). Thus the heat flux at the element boundary surface is specified as the sum of convective heat transfer terms based on Newton's law of cooling.

For the 1-D thermal energy equation, the element differential volume [consistent with Equation (3.22)] is defined as

$$d\Omega = \varepsilon A dx \qquad (4.4)$$

The differential surface area (for exchange of heat via Newton's law of cooling) is defined as

$$d\Gamma = P_h dx \qquad (4.5)$$

where P_h is the heated perimeter (i.e., the surface area exchanging heat). The heated and wetted perimeters can be different if desired.

It is now time to define the interpolation functions to be inserted into Equation (4.1). Recall that substitution of the interpolation (approximating) functions into the original equation results in an error called the residual; this residual multiplied by a weighting factor is then required to vanish in some average sense over the solution domain. The element fluid temperature is defined in terms of the interpolation functions as

$$T = \sum_{i=1}^{N_n} I_i T_i \qquad (4.6)$$

where I_i is the interpolation function for node i. In the current version of SAFSIM, the number of nodes (N_n) is 2 and the two interpolation functions are

$$I_1 = 1 - \frac{x}{L} \quad \text{and} \quad I_2 = \frac{x}{L} \qquad (4.7)$$

Thus it can be seen that for a two-noded 1-D element with $0 \le x \le L$, the element temperature is just a linear interpolation of the node temperatures:

$$T = \left(1 - \frac{x}{L}\right)T_1 + \frac{x}{L}T_2 \tag{4.8}$$

where T_1 and T_2 are the fluid temperatures at local nodes 1 and 2, respectively, of the finite element. Thus at the element center ($x = L/2$), the element fluid temperature is just the algebraic average of the node temperatures. A three-noded element would allow quadratic interpolation functions. Such higher-order elements can produce the same level of accuracy as a two-noded element with fewer elements. However, the amount of computations per element increases. In keeping with the engineering approach of SAFSIM, the simpler two-noded elements are employed; more elements can be used for increased resolution where necessary.

It is useful to define the temperature gradient in terms of the interpolation functions; thus

$$\frac{\partial T}{\partial x} = \sum_{i=1}^{N_n} B_i T_i \tag{4.9}$$

where B_i is the temperature gradient interpolation function defined as

$$B_i = \frac{\partial I_i}{\partial x} \tag{4.10}$$

Based on Equations (4.7) and (4.10):

$$B_1 = -\frac{1}{L} \quad \text{and} \quad B_2 = \frac{1}{L} \tag{4.11}$$

Also, the advective term is written in terms of the mass flow rate [based on Equation (3.4)] and the mass flow rate fractions [Equation (3.9)] as

$$c_p \frac{\partial(\rho v T)}{\partial x} = \left(\frac{\dot{m} c_p}{\varepsilon A}\right) \sum_{i=1}^{N_n} \frac{\partial N_i}{\partial x} \chi_i T_i \tag{4.12}$$

It is also convenient to write the interpolation functions in matrix/vector format; thus Equations (4.6) and (4.9) become, respectively,

$$T = \sum_{i=1}^{N_n} I_i T_i = \lfloor I \rfloor \{T\} \tag{4.13}$$

and

$$\frac{\partial T}{\partial x} = \sum_{i=1}^{N_n} B_i T_i = \lfloor B \rfloor \{T\} \tag{4.14}$$

where the half brackets $\lfloor\ \rfloor$ indicate a row vector and the curly brackets { } indicate a column vector.

Using matrix/vector notation, Equation (4.1) is written in the following concise form:

$$[C]\left\{\frac{\partial T}{\partial t}\right\} + ([K_k]+[K_c]+[K_A])\{T\} = \{R_{q'''}\} + \{R_c\} \tag{4.15}$$

where [C] is the element capacitance matrix, [K_k] is the fluid conduction conductance matrix, [K_c] is the convection heat transfer conductance matrix, [K_A] is the advection conductance matrix, {$R_{q'''}$} is the volumetric heating load vector, and {R_c} is the load vector for convection to an exchange surface or surfaces. These matrices and vectors are expressed by the following integrals:

$$[C] = \int_{\Omega} \rho c_v \{W\}\lfloor I \rfloor d\Omega = \int_{0}^{L} \rho c_v \{W\}\lfloor I \rfloor \varepsilon A dx \tag{4.16}$$

$$[K_k] = \int_{\Omega} \xi k_t \{W\}\lfloor B \rfloor d\Omega = \int_{0}^{L} \xi k_t \{W\}\lfloor B \rfloor \varepsilon A dx \tag{4.17}$$

$$[K_c] = \int_{\Gamma} h_c \{W\}\lfloor I \rfloor d\Gamma = \int_{0}^{L} h_c \{W\}\lfloor I \rfloor P_h dx \tag{4.18}$$

$$[K_A] = \int_{\Omega} c_p \{W\}\left(\frac{\dot{m}}{\varepsilon A}\right)\lfloor \chi B \rfloor d\Omega = \int_{0}^{L} \dot{m} c_p \{W\}\lfloor \chi B \rfloor dx \tag{4.19}$$

$$\{R_{q'''}\} = \int_{\Omega} q'''\{W\} d\Omega = \int_{0}^{L} q'''\{W\} \varepsilon A dx \tag{4.20}$$

$$\{R_c\} = \int_{\Gamma} h_c T_w \{W\} d\Gamma = \int_{0}^{L} h_c T_w \{W\} P_h dx \tag{4.21}$$

Before the integrals can be evaluated, the weighting function, W_i, must be defined. A common selection of the weighting functions is known as the Bubnov-Galerkin formulation. In this formulation, the weighting functions are set equal to the interpolation functions and the resulting matrices [after performing the integrations of Equations (4.16) through (4.21)] are referred to as the conventional element matrices. Such a formulation is analogous to central differencing in the finite difference method. Another formulation, known as the Petrov-Galerkin formulation, is useful when the governing equations contain an advective term. With this formulation, the weighting functions are

$$\lfloor W \rfloor = \lfloor I \rfloor + \eta \lfloor G \rfloor \tag{4.22}$$

where η is referred to as the upwind factor and the upwind interpolation functions, G, are given as

$$\lfloor G \rfloor = \left\lfloor 3\left(\frac{x^2}{L^2} - \frac{x}{L}\right) \quad -3\left(\frac{x^2}{L^2} - \frac{x}{L}\right) \right\rfloor \tag{4.23}$$

Figure 3 is included to graphically demonstrate the interpolation and weighting functions. The sum of the conventional (I_i) and upstream (G_i) interpolation functions equals the weighting function (W_i) for perfect upwinding ($\eta = 1$).

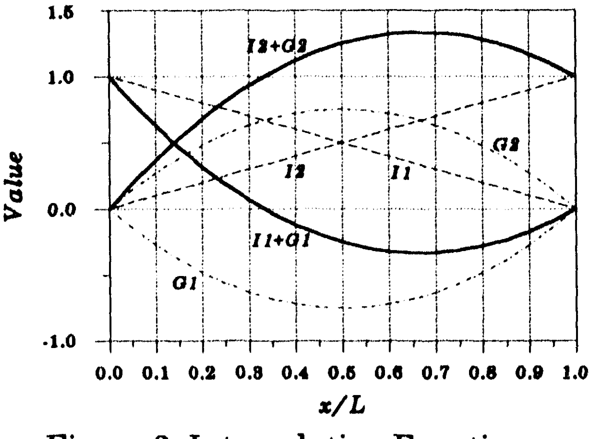


Figure 3. Interpolation Functions

When the upwind interpolation functions of Equation (4.23) are used, the resulting matrices are referred to as the upwind element matrices. Such a formulation is analogous to backward or upwind differencing in the finite difference method. The upwind concept is quite simple and merely reflects the fact that when the flow is advectively dominated, the dependent quantity (fluid temperature in this case) is primarily a function of the upwind conditions. On the other hand, when the flow is conductively dominated the dependent quantity is a function of both upwind and downwind conditions. The use of "upwinding" eliminates the spatial oscillations occurring in advective transport problems.

The upwind factor, η, ranges from 0 to 1, inclusive. If the upwind factor equals 0, the conventional element matrices result. However, based on accuracy considerations, oscillations in the solution occur if the grid Peclet number is greater than 2. The grid Peclet number is defined as

$$Pe_L = Re_L Pr_\xi = \frac{\dot{m} c_p L}{A \xi k_t} \tag{4.24}$$

where Re_L is the grid superficial Reynolds number based on the element length instead of the equivalent diameter, and Pr_ξ is the Prandtl number based on the dispersion-enhanced fluid thermal conductivity. The grid Peclet number characterizes the ratio of heat transfer by advection to the heat transfer by fluid conduction. Thus small grid Peclet numbers (less than about 2) indicate conductively (or diffusion) dominated flows while large grid Peclet numbers indicate advectively dominated flows. Any flow can be rendered conductively dominated by choosing small element lengths at the expense of computational efficiency. However, for many fluid flow problems of interest (especially those involving a gas), the element length would be prohibitively small.

Because the grid Peclet number is a function of mass flow rate, the element length required to make the flow conductively dominated changes as the flow rate changes. By introducing a variable upwind factor, it is possible to choose a value for each element based on the current value of the grid Peclet number. Thus the inclusion of an upwind element matrix offers a simple solution to the problems presented by an advective-conduction governing equation. The resulting hybrid finite element preserves oscillation-free solutions while maintaining the accuracy of the conventional matrix for conduction dominated flows. The major difficulty of the hybrid element is the loss of accuracy for flows that are neither conductively nor advectively dominated. More is provided on this later.

Jumping ahead somewhat, the inclusion of an upwind matrix for a steady-state advective-conduction flow problem (with no heat addition and with the assumed flow direction from node $i-1$ to node $i+1$) results in the following difference equation for three interior nodes after discretization and assembly:

$$\left[1 + Pe_L\left(\frac{\eta+1}{2}\right)\right]T_{i-1} - (2 + \eta Pe_L)T_i + \left[1 + Pe_L\left(\frac{\eta-1}{2}\right)\right]T_{i+1} = 0 \tag{4.25}$$

If Pe_L equals 0 (a pure conduction problem), the classic central differencing formula results in which a node fluid temperature is the algebraic average of the fluid temperatures of the adjacent nodes. If η is set to 1 (a pure advection problem indicative of large Pe_L), Equation (4.25) reduces to

$$T_i = \frac{(Pe_L + 1)T_{i-1} + T_{i+1}}{Pe_L + 2} \tag{4.26}$$

This equation shows that the larger the value of the grid Peclet number, the more heavily weighted is the upwind temperature. For example, for a grid Peclet number of 100, the upwind temperature has a weighting factor of 101/102 and the downwind temperature has a weighting factor of 1/102.

To ensure nonoscillatory flows when the grid Peclet number is greater than 2, it is necessary to select the upwind factor such that

$$\eta \geq 1 - \frac{2}{Pe_L} \tag{4.27}$$

Many schemes[19] have been proposed for choosing the upwind factor. However, because these schemes are based on an analytical solution for steady, constant-property flow with no source terms or convection heat transfer, they are not relevant for most practical flow situations. With a hybrid formulation, accuracy is diminished for upwind factors between 0 and 1 because the upwind element component serves as a source of numerical diffusion. Also, the inclusion of source terms exacerbates the problem. Therefore, the following simple scheme is implemented in SAFSIM:

$$\eta = 0 \text{ for } Pe_L \leq 1,\ \eta = \tfrac{1}{2}(Pe_L - 1) \text{ for } 1 < Pe_L < 3, \text{ and } \eta = 1 \text{ for } Pe_L \geq 3 \tag{4.28}$$

Thus except for Pe_L between 1 and 3, the upwind factor is either 0 (conductively dominated flow) or 1 (advectively dominated flow). This scheme imposes full upwinding when the grid Peclet number reaches the relatively low value of 3. Thus advection dominated solutions are emphasized at the expense of small conduction related errors. As the grid Peclet number increases, these small errors grow even smaller. Likewise, conduction dominated solutions are emphasized at the expense of small advection related errors.

Assigning an upwind factor for each finite element allows reasonable flow solutions to be calculated when part of a flow system is advectively dominated while another part is conductively dominated. Also, for flow in an element that is neither conductively nor advectively dominated, an upwind factor between 0 and 1 reflects an averaging of the two concurrent processes. However, accuracy degradation may become significant for these flows.

Much controversy exists in the finite element community with regard to upwind elements. Although upwind elements eliminate spatial oscillation, they introduce some degree of numerical diffusion. Many contend that the oscillations are indicative of too coarse of a mesh or the use of inappropriate boundary conditions. On the other hand, their use allows oscillation-free solutions for any given mesh. Because the use of a coarse mesh is desirable in complicated system level problems (to increase execution speed), the hybrid conventional/upwind formulation is implemented in SAFSIM.

Now that the interpolation and weighting functions have been specified, the integrals of Equations (4.16) through (4.21) can be performed. The integrations result in the following expressions for the capacitance and conductance matrices:

$$[C]=\frac{\rho c_v V}{6}\begin{bmatrix}2 & 1\\ 1 & 2\end{bmatrix}+\eta\frac{\rho c_v V}{4}\begin{bmatrix}-1 & -1\\ 1 & 1\end{bmatrix} \tag{4.29}$$

$$[K_A]=\frac{\dot{m}c_p}{2}\begin{bmatrix}-\chi_1 & \chi_2\\ -\chi_1 & \chi_2\end{bmatrix}+\eta\frac{\dot{m}c_p}{2}\begin{bmatrix}\chi_1 & -\chi_2\\ -\chi_1 & \chi_2\end{bmatrix} \tag{4.30}$$

$$[K_k]=\frac{\xi k_t \varepsilon A}{L}\begin{bmatrix}1 & -1\\ -1 & 1\end{bmatrix} \tag{4.31}$$

$$[K_c]=\frac{h_c A_w}{6}\begin{bmatrix}2 & 1\\ 1 & 2\end{bmatrix}+\eta\frac{h_c A_w}{4}\begin{bmatrix}-1 & -1\\ 1 & 1\end{bmatrix} \tag{4.32}$$

$$\{R_c\}=\frac{h_c A_w T_w}{2}\begin{Bmatrix}1\\ 1\end{Bmatrix}+\eta\frac{h_c A_w T_w}{2}\begin{Bmatrix}-1\\ 1\end{Bmatrix} \tag{4.33}$$

$$\{R_{q'''}\}=\frac{q_t''' V}{2}\begin{Bmatrix}1\\ 1\end{Bmatrix}+\eta\frac{q_t''' V}{2}\begin{Bmatrix}-1\\ 1\end{Bmatrix} \tag{4.34}$$

Except for the conduction term, each matrix is composed of a conventional element matrix contribution (analogous to central differencing) plus the upwind factor multiplied by an upwind element matrix contribution (analogous to upwind differencing).

It should be noted that these matrices are for positive fluid flow defined as flow from local node 1 to local node 2 of the element. Similar matrices result for reverse flow. Because the analyst is allowed to define the local nodes as desired and to connect elements together arbitrarily (to enhance versatility), much bookkeeping is required to properly assemble the element matrices into a global matrix. However, this bookkeeping is transparent to the analyst.

An unfortunate property of the selected hybrid finite element is that it does not result in perfect upwinding when η equals 1, as shown by Equation (4.26). This can also be demonstrated by setting the upwind factor to 1 in Equation (4.29), the capacitance matrix, and in Equation (4.32), the convection heat transfer conductance matrix. For example, Equation (4.32) simplifies to:

$$[K_c] = h_c A_w \begin{bmatrix} 1/12 & -1/12 \\ 5/12 & 7/12 \end{bmatrix} \qquad (4.35)$$

Perfect upwinding requires that the entries in the top row of this matrix equal zero. It is possible to achieve perfect upwinding if the conventional element component of Equations (4.29) and (4.32) are appropriately modified. One such modification for achieving perfect upwinding is to replace the consistent matrix with what can be referred to as a lumped matrix. (The conventional components are expressed in what is sometimes called the consistent form because they are consistent with the derivations of the general finite element equations.) Thus Equations (4.29) and (4.32) are now replaced (temporarily) with the following:

$$[C] = \frac{\rho c_v V}{4}\begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix} + \eta \frac{\rho c_v V}{4}\begin{bmatrix} -1 & -1 \\ 1 & 1 \end{bmatrix} \qquad (4.36)$$

for the capacitance matrix, and

$$[K_c] = \frac{h_c A_w}{4}\begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix} + \eta \frac{h_c A_w}{4}\begin{bmatrix} -1 & -1 \\ 1 & 1 \end{bmatrix} \qquad (4.37)$$

for the convection heat transfer conductance matrix. Unlike the consistent matrix, the lumped matrix applies equal weighting to both nodes of the element thus allowing perfect upwinding when the upwind factor equals 1.

The matrices of Equations (4.36) and (4.37) are not the final form used in SAFSIM because an improvement in accuracy can be achieved by making an additional modification. This modification allows the accuracy of a higher-order element to be achieved with a simpler linear element. To illustrate the need for further modification, it is useful to present the following simplified thermal energy equation:

$$\dot{m} c_p (T_2 - T_1) = h_c A_w (T_w - T) \qquad (4.38)$$

This is an integral steady-state energy balance for a 1-D control volume (finite element) with conduction and volumetric heat sources omitted for simplicity.

Assuming that T_1 is a specified boundary condition, it is necessary to express T (an average fluid temperature for the finite element) as a function of T_1 and T_2 in order to solve for T_2. For a two-noded element, a linear relationship is required of the form:

$$T = w_c T_1 + (1 - w_c) T_2 \tag{4.39}$$

where w_c is a weighting factor between 0 and 1. Equations (4.38) and (4.39) can be combined to yield:

$$T_2 = \frac{\left(\frac{h_c A_w}{\dot{m} c_p}\right) T_w + T_1 \left[1 - w_c \left(\frac{h_c A_w}{\dot{m} c_p}\right)\right]}{1 + (1 - w_c)\left(\frac{h_c A_w}{\dot{m} c_p}\right)} \tag{4.40}$$

The quantity in parentheses can be expressed in terms of the Stanton number:

$$\left(\frac{h_c A_w}{\dot{m} c_p}\right) = \left(\frac{h_c}{\rho v c_p}\right)\left(\frac{A_w}{\varepsilon A}\right) = St\left(\frac{A_w}{\varepsilon A}\right) = \tilde{St} \tag{4.41}$$

where $\tilde{St}$ is referred to as a modified Stanton number. The Stanton number is a common dimensionless parameter defined as the ratio of the heat transfer by convection at the wall to the heat transfer by advection. For T_1 less than T_w, inspection of Equation (4.40) reveals the nonphysical result that T_2 is greater than T_w whenever the modified Stanton number is greater than the inverse of the weighting factor. Similarly, for T_1 greater than T_w, T_2 is less than T_w whenever the modified Stanton number is greater than the inverse of the weighting factor. Stated mathematically:

$$T_2 > T_w \quad \text{for} \quad T_1 < T_w \quad \text{when} \quad \tilde{St} > \frac{1}{w_c} \tag{4.42}$$

and

$$T_2 < T_w \quad \text{for} \quad T_1 > T_w \quad \text{when} \quad \tilde{St} > \frac{1}{w_c} \tag{4.43}$$

For example, if the weighting factor is given the usual value of 1/2, the critical modified Stanton number ($1/w_c$) equals 2. Then, for T_1 less than T_w, if $\tilde{St}$ is greater than 2, T_2 is greater than the wall temperature. The reason for the nonphysical result is that a linear fluid temperature profile is employed when in fact the profile is exponential.

The problem can be avoided by using more elements of smaller size. Thus the wall exchange area for the element is reduced thereby reducing the modified Stanton number. With this approach, the exponential profile is approximated with multiple linear profiles, which of course is the essence of any numerical approximating procedure. The problem is that the modified Stanton number is a function of the mass flow rate, which can vary during a simulation. Therefore it is not possible to *a priori* select an appropriate finite element length to ensure the modified Stanton number remains below the critical value. This is particularly a problem for laminar and transition flow regimes where the heat transfer coefficient is not a strong function of mass flow rate. However, a problem can also arise in turbulent flow, especially for flow in porous media where the exchange surface area can be very large for a given finite element volume.

A common solution to the problem is to use a weighting factor of 0, which is equivalent to setting T equal to T_2. This avoids the nonphysical result but usually introduces gross inaccuracies. A solution to the problem can be achieved by proper choice of the weighting factor based on the flow conditions of the finite element. The procedure for selecting the appropriate weighting factor begins with the solution to the following 1-D equation:

$$\dot{m}c_p \frac{dT}{dx} = h_c P_h (T_w - T) + q_t''' \varepsilon A \tag{4.44}$$

This equation is equivalent to Equation (4.38) written in differential form and includes volumetric heating for increased generality. Assuming that T_w and h_c are uniform (not functions of x), and that T equals T_1 at $x = 0$, the solution to this equation is

$$T(x) = T_w + \frac{q_t''' \varepsilon A}{h_c P_h} + \left(T_1 - T_w - \frac{q_t''' \varepsilon A}{h_c P_h} \right) \exp\left[-\left(\frac{h_c P_h}{\dot{m} c_p} \right) x \right] \tag{4.45}$$

This solution demonstrates the exponential nature of the combined advective-convection heat transfer problem. At $x = L$, with $A_w = P_h L$ and $q_t = q_t''' \varepsilon A L$, this equation provides the following relation for T_2:

$$T_2 = T_w + \frac{q_t}{h_c A_w} + \left(T_1 - T_w - \frac{q_t}{h_c A_w} \right) \exp\left[-\left(\frac{h_c A_w}{\dot{m} c_p} \right) \right] \tag{4.46}$$

Taking the natural logarithm of this equation produces

$$\left(\frac{h_c A_w}{\dot{m} c_p}\right) = \widetilde{St} = -\ln\left(\frac{T_2 - T_w - \dfrac{q_t}{h_c A_w}}{T_1 - T_w - \dfrac{q_t}{h_c A_w}}\right) \qquad (4.47)$$

where the modified Stanton number was introduced in Equation (4.41). The next step is the integration of Equation (4.44) over the flow length; upon rearrangement, this equation can be written as

$$\overline{(T_w - T)} = \frac{(T_2 - T_1)}{\widetilde{St}} - \frac{q_t}{h_c A_w} \qquad (4.48)$$

where $\overline{(T_w - T)}$ is the average temperature difference to be used in Newton's law of cooling. Inserting Equation (4.47) into Equation (4.48) provides the following expression for the average temperature difference:

$$\overline{(T_w - T)} = \Delta T_{LM} = \frac{T_1 - T_2}{\ln\left[\dfrac{T_2 - T_w - \dfrac{q_t}{h_c A_w}}{T_1 - T_w - \dfrac{q_t}{h_c A_w}}\right]} - \frac{q_t}{h_c A_w} \qquad (4.49)$$

where it is seen that the appropriate average temperature difference to use is the log-mean temperature difference, ΔT_{LM}, familiar from heat exchanger theory but complicated somewhat by the addition of the direct heating term.

The proper average value of T to use in Newton's law of cooling is simply

$$T = T_w - \Delta T_{LM} \qquad (4.50)$$

where the log-mean temperature difference is provided by Equation (4.49). Equation (4.39) can be written in terms of w_c as follows:

$$w_c = \frac{T - T_2}{T_1 - T_2} \qquad (4.51)$$

where w_c is now defined as the log-mean weighting factor for convection heat transfer. Combining Equations (4.50) and (4.51) gives

$$w_c = \frac{(T_w - \Delta T_{LM}) - T_2}{T_1 - T_2} \tag{4.52}$$

Also, Equation (4.46) is written here in terms of the modified Stanton number:

$$T_2 = T_w + \frac{q_t}{h_c A_w} + \left(T_1 - T_w - \frac{q_t}{h_c A_w} \right) \exp\left[-\tilde{St} \right] \tag{4.53}$$

Thus the appropriate value of w_c can be determined by the solution of Equations (4.49), (4.52), and (4.53). Upon first inspection, it appears that the log-mean weighting factor is a function of T_1, T_2, T_w, $q_t/(h_c A_w)$, and $\tilde{St}$. Fortuitously, although not obvious, the weighting factor is only a function of the modified Stanton number; the other parameters serve only to translate the exponential fluid temperature profile and do not alter its shape. Thus the log-mean weighting factor can be calculated as a function of the modified Stanton number, independent of T_1, T_2, T_w, and $q_t /(h_c A_w)$. Figure 4 presents the results of this calculation and a curve fit of the log-mean weighting factor data (as used in SAFSIM) is provided by the following equation:

$$w_c = \frac{0.500068210 + 0.039853698 \cdot \tilde{St} + 0.000416401 \cdot \tilde{St}^2}{1 + 0.23517031 \cdot \tilde{St} + 0.059687353 \cdot \tilde{St}^2} \tag{4.54}$$

For any value of the modified Stanton number, the appropriate log-mean weighting factor can be calculated by Equation (4.54). Thus *a priori* determination of w_c allows a two-noded linear finite element to approximate the accuracy of a higher-order element with only minimal computational overhead. Equation (4.37) is now written to take advantage of this weighting factor. Thus

$$[K_c] = \frac{h_c A_w}{2} \begin{bmatrix} (1 - w_c) & w_c \\ w_c & (1 - w_c) \end{bmatrix} + \eta \frac{h_c A_w}{2} \begin{bmatrix} -(1 - w_c) & -w_c \\ w_c & (1 - w_c) \end{bmatrix} \tag{4.55}$$

Now the convective heat transfer conductance matrix contains variable coefficients. If w_c happens to equal 1/2, Equation (4.55) provides the lumped form of the convection heat transfer conductance matrix [Equation (4.37)], and if w_c equals 1/3, it provides the consistent form of the convection heat transfer conductance matrix [Equation (4.32)]. Of course, w_c can take on any value between 0 and 1/2, as dictated by the flow conditions.

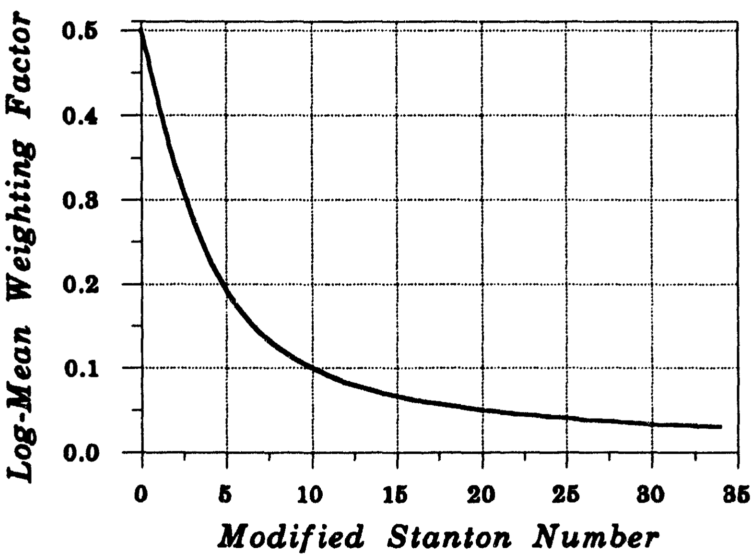


Figure 4. The Log-Mean Weighting Factor

The advective-conduction oscillatory problem is addressed with the use of a grid-Peclet-number-based upwind factor and the advective-convection oscillatory problem is addressed with the use of a modified-Stanton-number-based log-mean weighting factor. One additional spatial oscillatory problem arises due to the advective-capacitance nature of the thermal energy equation. This problem arises whenever a "traveling wave" phenomena is numerically modeled using an Eulerian (or fixed) grid.

Equation (4.38) was presented as a simplified thermal energy equation to illustrate the basic advective-convection problem. Similarly, a simplified thermal energy equation is written here to illustrate the advective-capacitance problem. Thus

$$\dot{m}c_p\left(T_2^{n+1} - T_1^{n+1}\right) = -\rho c_v V\left(\frac{T^{n+1} - T^n}{\Delta t_f}\right) \qquad (4.56)$$

where the superscripts n+1 and n refer to the future and current time values, respectively, and Δt_f is the fluid mechanics time step. This simple equation for a 1-D finite element balances the advective transfer of thermal energy (in terms of fluid temperature) with the time rate of change of temperature. A simple finite differencing in time is used for the dynamic (time-derivative) term and the advective term is written in terms of the future values of the node temperatures (i.e., implicitly). Once again, the finite element fluid temperature at any time can be written in terms of the two node temperatures and a dynamic weighting factor, w_d. Thus

$$T^{n+1} = w_d T_1^{n+1} + (1 - w_d) T_2^{n+1} \quad \text{or} \quad T^n = w_d T_1^n + (1 - w_d) T_2^n \tag{4.57}$$

Combining Equations (4.56) and (4.57) produces

$$T_2^{n+1} = \frac{\left(\dfrac{\rho c_v V}{\dot{m} c_p \Delta t_f}\right) T^n + T_1^{n+1}\left[1 - w_d\left(\dfrac{\rho c_v V}{\dot{m} c_p \Delta t_f}\right)\right]}{1 + (1 - w_d)\left(\dfrac{\rho c_v V}{\dot{m} c_p \Delta t_f}\right)} \tag{4.58}$$

Comparison of this equation with Equation (4.40) reveals their similarity, with the quantity in parentheses serving the same role as the modified Stanton number in the advective-convection equation. Rearrangement of this quantity provides:

$$\frac{\rho c_v V}{\dot{m} c_p \Delta t_f} = \frac{c_v}{c_p}\left(\frac{L}{v \Delta t_f}\right) = (\gamma Co)^{-1} \tag{4.59}$$

where γ is the ratio of specific heats and Co is a material Courant number. This number is familiar from the well-known Courant condition for 1-D explicit numerical formulations of a time-dependent advective equation. Based on stability considerations, the Courant condition requires that

$$Co = \frac{v \Delta t_f}{L} < 1 \tag{4.60}$$

Analogous once again to the advective-convection equation, because of the assumed linear fluid temperature profile, Equation (4.58) provides nonphysical fluid temperature results according to the following criteria:

$$T_2^{n+1} > T^n \quad \text{for} \quad T_1^{n+1} < T^n \quad \text{when} \quad (\gamma Co)^{-1} > \frac{1}{w_d} \tag{4.61}$$

and

$$T_2^{n+1} < T^n \quad \text{for} \quad T_1^{n+1} > T^n \quad \text{when} \quad (\gamma Co)^{-1} > \frac{1}{w_d} \tag{4.62}$$

Again, T_1 is the assumed boundary condition. If γCo is considered as a modified material Courant number, then nonphysical results occur when the inverse of the modified Courant number is greater than a critical value given by $1/w_d$. Restated mathematically, to prevent nonphysical results requires that

$$Co > \frac{w_d}{\gamma} \tag{4.63}$$

Thus for a given element length and fluid velocity, this criteria places a minimum acceptable-value restriction on the time step. This is in contrast to the Courant condition which places a maximum acceptable-value restriction. For example, if w_d equals 1/2 and γ equals 5/4, a large enough time step must be selected such that the Courant number is greater than 2/5. Although an explicit formulation of the thermal energy equation is not used in SAFSIM, the comparison to the Courant condition is presented as a point of interest. The restriction of Equation (4.63) is related to the advection of energy in time with respect to the dynamic fluid temperature term and is independent of the level of implicitness assumed for the spatial terms.

Again, the difficulty is associated with a traveling wave front and is a classic problem associated with the numerical modeling of advective phenomena based on a fixed point of reference. Many solutions to this problem have been proposed in the finite difference community and are migrating to the finite element community. The solutions involve the use of hybrid Eulerian-Lagrangian elements or higher-order elements, which are, of course, more complicated than a simple linear element. One of these solutions may be incorporated into a later version of SAFSIM; however, the current version of SAFSIM minimizes the problem by an appropriate selection of the weighting factor, w_d. Based on the similarity of Equations (4.58) and (4.40), some type of log-mean weighting factor is indicated. The weighting factor equation developed for the advective-convection problem [Equation (4.54)] with the inverse modified Courant number used in place of the modified Stanton number has been found to prevent the nonphysical results associated with the time-dependent advection problem. Thus using the curve fit of Equation (4.54) provides

$$w_d = \frac{0.500068210 + 0.039853698 \cdot (\gamma Co)^{-1} + 0.000416401 \cdot (\gamma Co)^{-2}}{1 + 0.23517031 \cdot (\gamma Co)^{-1} + 0.059687353 \cdot (\gamma Co)^{-2}} \tag{4.64}$$

Although a linear element suffers from numerical diffusion (which can be reduced by decreasing the element length) for problems involving a traveling wave front, the use of the dynamic log-mean weighting factor, w_d, is simple and prevents spatial oscillations while removing the associated time step restriction. Introducing the weighting factor into Equation (4.36), the new capacitance matrix becomes

$$[C] = \frac{\rho c_v V}{2}\begin{bmatrix}(1-w_d) & w_d \\ w_d & (1-w_d)\end{bmatrix} + \eta\frac{\rho c_v V}{2}\begin{bmatrix}-(1-w_d) & -w_d \\ w_d & (1-w_d)\end{bmatrix} \tag{4.65}$$

The development of the finite element conductance and capacitance matrices and load vectors is now complete. These matrices and vectors are repeated here for convenience with the conventional and upwind matrices added together for conciseness. Thus

$$[C] = \frac{\rho c_v V}{2}\begin{bmatrix}(1-\eta)(1-w_d) & (1-\eta)w_d \\ (1+\eta)w_d & (1+\eta)(1-w_d)\end{bmatrix} \tag{4.66}$$

$$[K_A] = \frac{\dot{m}c_p}{2}\begin{bmatrix}-(1-\eta)\chi_1 & (1-\eta)\chi_2 \\ -(1+\eta)\chi_1 & (1+\eta)\chi_2\end{bmatrix} \tag{4.67}$$

$$[K_k] = \frac{\xi k_t \varepsilon A}{L}\begin{bmatrix}1 & -1 \\ -1 & 1\end{bmatrix} \tag{4.68}$$

$$[K_c] = \frac{h_c A_w}{2}\begin{bmatrix}(1-\eta)(1-w_c) & (1-\eta)w_c \\ (1+\eta)w_c & (1+\eta)(1-w_c)\end{bmatrix} \tag{4.69}$$

$$\{R_c\} = \frac{h_c A_w T_w}{2}\begin{Bmatrix}(1-\eta) \\ (1+\eta)\end{Bmatrix} \tag{4.70}$$

$$\{R_{q'''}\} = \frac{q_t}{2}\begin{Bmatrix}(1-\eta) \\ (1+\eta)\end{Bmatrix} \tag{4.71}$$

The final step is to define the level of implicitness for the spatial terms in Equation (4.15). In other words, the fluid temperatures in the $\{T\}$ vector must be evaluated at the current or future time, or some combination of the two. Using a simple finite difference formulation for the time-derivative capacitance term and defining θ_{FM} as the fluid mechanics implicitness factor, Equation (4.15) can be written as

$$[C]\left\{\frac{T^{n+1}-T^n}{\Delta t_f}\right\} + [K_t]\{T\}^{\theta_{FM}} = \{R_t\}^{\theta_{FM}} \tag{4.72}$$

where the total conductance matrix $[K_t]$ is

$$[K_t]=[K_k]+[K_c]+[K_A] \tag{4.73}$$

and the total load vector $\{R_t\}$ is

$$\{R_t\}=\{R_{q'''}\}+\{R_c\} \tag{4.74}$$

Also, the implicitness factor is defined such that

$$t^{\theta_{FM}} = t^n + \theta_{FM}\Delta t_f \tag{4.75}$$

and $0 \le \theta_{FM} \le 1$. The following definitions can now be made:

$$\{T\}^{\theta_{FM}} = (1-\theta_{FM})\{T\}^n + \theta_{FM}\{T\}^{n+1} \tag{4.76}$$

and

$$\{R_t\}^{\theta_{FM}} = (1-\theta_{FM})\{R_t\}^n + \theta_{FM}\{R_t\}^{n+1} \tag{4.77}$$

Now, substitution of Equations (4.76) and (4.77) into Equation (4.72) gives

$$\begin{aligned}&\left[\frac{1}{\Delta t_f}[C]+\theta_{FM}[K_t]\right]\{T\}^{n+1} = \\ &\left[\frac{1}{\Delta t_f}[C]-(1-\theta_{FM})[K_t]\right]\{T\}^n + (1-\theta_{FM})\{R_t\}^n + \theta_{FM}\{R_t\}^{n+1}\end{aligned} \tag{4.78}$$

This can be expressed more concisely using the following definitions:

$$[\overline{K}_t] = \frac{1}{\Delta t_f}[C]+\theta_{FM}[K_t] \tag{4.79}$$

and

$$\{\overline{R}_t\} = \left[\frac{1}{\Delta t_f}[C]-(1-\theta_{FM})[K_t]\right]\{T\}^n + (1-\theta_{FM})\{R_t\}^n + \theta_{FM}\{R_t\}^{n+1} \tag{4.80}$$

where $[\overline{K}_t]$ and $\{\overline{R}_t\}$ are the effective total conductance matrix and the effective total load vector, respectively. Now Equation (4.78) can be written as

$$[\overline{K}_t]\{T\}^{n+1} = \{\overline{R}_t\} \tag{4.81}$$

Written in this form, the level of implicitness can be selected by the choice of the implicitness factor, θ_{FM}. The four most common choices for θ_{FM} are 0, 1/2, 2/3, and 1. A value of 0 results in a fully explicit formulation which suffers from the time step restriction presented in Equation (4.60), i.e., the Courant condition. An additional restriction arises due to the inclusion of the fluid conduction term. This restriction is known as the diffusion condition and in 1-D is given as

$$\Delta t_f < \frac{\rho c_p L^2}{2k_t} \tag{4.82}$$

Although the fully explicit formulation is easy to program and computationally efficient, the two potential time step restrictions are undesirable in a general-purpose system program such as SAFSIM.

Choosing an implicitness factor of 1/2 results in what is commonly referred to as the Crank-Nicolson formulation. And a factor of 2/3 results in the Galerkin formulation (the 2/3 number arises in the consistent form of the finite element matrix when time is treated as an additional dimension). Both the Crank-Nicolson and Galerkin formulations are unconditionally stable but may exhibit an oscillatory transient response if the time step is too large. The current version of SAFSIM employs a fully implicit formulation (θ_{FM} = 1) for solution of the thermal energy equation. The fully implicit formulation is unconditionally stable and does not suffer an oscillatory response for any time step size, at the expense of a small decrease in accuracy compared to Crank-Nicolson and Galerkin formulations. The fully implicit formulation is also relatively easy to program and computationally efficient (fast), making it an appealing choice for SAFSIM. Future versions of SAFSIM may incorporate an option to allow the user to select the level of implicitness via the input file if the need arises.

The solution procedure and numerical methods for the thermal energy equation are discussed in detail after development of the finite element matrices for the mechanical energy and mass fraction equations.

4.2 Mechanical Energy Equation

As mentioned in Section 4, the finite element equations for the mechanical energy equation are derived using a combination of the direct and integral balance approaches. The thermal energy equation contains a single dependent variable, namely fluid temperature T. The mechanical energy equation, however, contains two dependent variables: pressure p, and mass flow rate $\dot{m}$. An iterative approach is employed to resolve this problem and is discussed in Section 6 where the numerical solution methods are explained.

Use of the direct and integral balance approaches simplifies the derivation of the finite element equations because the integral balance equations derived in Section 3 serve as the starting point. The control volume now becomes the finite element and the ports are the local nodes. Specifically, Equation (3.47) provides the basic relationship to be employed in the finite element derivation and is repeated here for convenience:

$$F(\tilde{p}_2 - \tilde{p}_1) = \dot{m} - Fg_z\Delta z - \frac{FL}{\rho \varepsilon A}\left(\frac{\partial \dot{m}}{\partial t} - \frac{\dot{m}}{2\rho}\frac{\partial \rho}{\partial t}\right) \tag{3.47}$$

The first step is to write the conservation of mass for an element in terms of the node mass flow rates and source terms. The conservation of mass governing equations are developed in Section 3. Combining Equations (3.7) and (3.9) and rearranging produces

$$\dot{m} = \dot{m}_1 + (1 - w_m)\left(S - V\frac{\partial \rho}{\partial t}\right) \tag{4.83}$$

for the first local node and

$$\dot{m} = \dot{m}_2 - w_m\left(S - V\frac{\partial \rho}{\partial t}\right) \tag{4.84}$$

for the second local node of the element.

Now the mechanical energy equation [given by Equation (3.47)] is combined with each of the mass continuity relationships [Equations (4.83) and (4.84)] to produce the following equations for the two nodes:

$$\begin{gathered} F\left(\frac{\chi_1}{\rho_1}p_1 - \frac{\chi_2}{\rho_2}p_2\right) = \\ -\dot{m}_1 - (1 - w_m)\left(S - V\frac{\partial \rho}{\partial t}\right) + Fg_z\Delta z + F\frac{L}{\rho \varepsilon A}\left(\frac{\partial \dot{m}}{\partial t} - \frac{\dot{m}}{2\rho}\frac{\partial \rho}{\partial t}\right) \end{gathered} \tag{4.85}$$

and

$$-F\left(\frac{\chi_1}{\rho_1}p_1-\frac{\chi_2}{\rho_2}p_2\right)= \\ \dot{m}_2-w_m\left(S-V\frac{\partial\rho}{\partial t}\right)-Fg_z\Delta z-F\frac{L}{\rho\varepsilon A}\left(\frac{\partial\dot{m}}{\partial t}-\frac{\dot{m}}{2\rho}\frac{\partial\rho}{\partial t}\right) \tag{4.86}$$

The advantage of this approach is that mass continuity, enforced at the nodes, is included directly into the mechanical energy equation. The values for χ_1 and χ_2 are determined by solution of Equations (3.9), which are forms of the mass conservation equation. Thus solution of Equations (4.85) and (4.86) automatically satisfies conservation of mass. Another advantage of this approach is that the node mass flow rates appear as natural boundary conditions. Equations (4.85) and (4.86) are now written in matrix form as

$$[F]\{p\}=\{M_e\} \tag{4.87}$$

where [*F*] is the fluidity matrix given by

$$[F]=F\begin{bmatrix}\left(\chi_1/\rho_1\right) & -\left(\chi_2/\rho_2\right)\\ -\left(\chi_1/\rho_1\right) & \left(\chi_2/\rho_2\right)\end{bmatrix} \tag{4.88}$$

$\{p\}$ is the node pressure vector,

$$\{p\}=\begin{Bmatrix}p_1\\ p_2\end{Bmatrix} \tag{4.89}$$

and $\{M_e\}$ is the effective mass flow rate load vector given as the sum of the node mass flow rate load vector $\{\dot{m}\}$, the effective fluid mass source load vector $\{S_e\}$, the acceleration load vector $\{g\}$, and the dynamic mass flow rate load vector $\{M_D\}$. Thus

$$\{M_e\}=\{\dot{m}\}+\{S_e\}+\{g\}+\{M_D\} \tag{4.90}$$

where

$$\{\dot{m}\}=\begin{Bmatrix}-\dot{m}_1\\ \dot{m}_2\end{Bmatrix} \tag{4.91}$$

$$\{S_e\} = \left(S - V\frac{\partial p}{\partial t}\right)\begin{Bmatrix} -(1 - w_m) \\ -w_m \end{Bmatrix} \tag{4.92}$$

$$\{g\} = Fg_z\Delta z\begin{Bmatrix} 1 \\ -1 \end{Bmatrix} \tag{4.93}$$

$$\{M_D\} = F\frac{L}{\rho\varepsilon A}\left(\frac{\partial \dot{m}}{\partial t} - \frac{\dot{m}}{2\rho}\frac{\partial p}{\partial t}\right)\begin{Bmatrix} 1 \\ -1 \end{Bmatrix} \tag{4.94}$$

When the local load vectors are assembled to form a global load vector, the mass flow rates for internal nodes cancel except when specified as a boundary condition. Thus, if the element flow rates are assumed known, Equation (4.87) can be solved for the node pressures. With the pressures calculated, the element mass flow rates can be determined using the following relationship between pressure drop and mass flow rate:

$$\dot{m}^{n+1} = \frac{\left(\frac{\chi_2}{\rho_2}p_2 - \frac{\chi_1}{\rho_1}p_1\right)^{n+1} + g_z\Delta z - \frac{L}{\rho^{n+1}\varepsilon A\Delta t_f}\dot{m}^n}{\Re^{n+1} - \frac{L}{2\rho^{n+1}\varepsilon A\Delta t_f}\left(1 + \frac{\rho^n}{\rho^{n+1}}\right)} \tag{4.95}$$

where $\Re$ is the element resistance [defined in Equation (3.46)], which is the inverse of the fluidity F. This equation is simply Equation (3.47) written in terms of mass flow rate, adopting a fully implicit formulation consistent with the thermal energy equation ($\theta_{FM} = 1$). Thus the spatial terms and fluid properties are evaluated at the future time and a simple finite difference of the time-derivative terms is used. Because the calculated pressures are based on assumed values of the mass flow rates, an iterative procedure is required to determine the pressures and corresponding mass flow rates. The iterative process used between pressure and mass flow rate is discussed more fully later.

An interesting feature of Equation (4.87) is that in contrast to the thermal energy equation, there is only a single coefficient matrix, namely the fluidity matrix. Also, the potential energy and dynamic terms of the governing equation appear as part of the load vector.

Inspection of the fluidity matrix indicates that the equation for pressure is diffusive in nature. This is indicated by the symmetric nature of the matrix with the offdiagonal terms of opposite sign than the diagonal terms, similar to the conduction matrix of the thermal energy equation [Equation (4.68)]. This

indicates that Equation (4.87) is useful only for subsonic flows. However, like the thermal energy equation, the mechanical energy equation is an advective-diffusion equation when both subsonic and supersonic flows are considered. To allow the treatment of supersonic flows, it is necessary to modify Equation (4.87) to include an advective component by the inclusion of an upwind element component.✦

By analogy to the thermal energy equation, the finite element equation for mechanical energy can be modified to allow supersonic flow predictions by the addition of an advective upwind matrix component; thus

$$[F] = F\begin{bmatrix} \left(\chi_1/\rho_1\right) & -\left(\chi_2/\rho_2\right) \\ -\left(\chi_1/\rho_1\right) & \left(\chi_2/\rho_2\right) \end{bmatrix} + \eta_F F\begin{bmatrix} -\left(\chi_1/\rho_1\right) & \left(\chi_2/\rho_2\right) \\ -\left(\chi_1/\rho_1\right) & \left(\chi_2/\rho_2\right) \end{bmatrix} \tag{4.96}$$

and

$$\{\dot{m}\} = \begin{Bmatrix} -\dot{m}_1 \\ \dot{m}_2 \end{Bmatrix} + \eta_F \begin{Bmatrix} \dot{m}_1 \\ \dot{m}_2 \end{Bmatrix} \tag{4.97}$$

$$\{S_e\} = \left(S - V\frac{\partial p}{\partial t}\right)\begin{Bmatrix} -(1-w_m) \\ -w_m \end{Bmatrix} + \eta_F\left(S - V\frac{\partial p}{\partial t}\right)\begin{Bmatrix} (1-w_m) \\ -w_m \end{Bmatrix} \tag{4.98}$$

$$\{g\} = Fg_z\Delta z\begin{Bmatrix} 1 \\ -1 \end{Bmatrix} + \eta_F Fg_z\Delta z\begin{Bmatrix} -1 \\ -1 \end{Bmatrix} \tag{4.99}$$

$$\{M_D\} = F\frac{L}{\rho\varepsilon A}\left(\frac{\partial \dot{m}}{\partial t} - \frac{\dot{m}}{2\rho}\frac{\partial \rho}{\partial t}\right)\begin{Bmatrix} 1 \\ -1 \end{Bmatrix} + \eta_F F\frac{L}{\rho\varepsilon A}\left(\frac{\partial \dot{m}}{\partial t} - \frac{\dot{m}}{2\rho}\frac{\partial \rho}{\partial t}\right)\begin{Bmatrix} -1 \\ -1 \end{Bmatrix} \tag{4.100}$$

where η_F is the upwind factor for the mechanical energy equation. As was noted for the thermal energy equation, these matrices are for positive flow defined from local node 1 to local node 2; similar matrices result for reverse flow.

✦ Although upwind elements for the mechanical energy equation are not fully implemented in the current version of SAFSIM, preliminary results with an experimental version are encouraging. A brief description of upwind elements is included here for completeness.

When the flow is subsonic, the resulting potential flow equations are elliptic in the space coordinate (parabolic in time) and the pressure at a node is a function of both upwind and downwind conditions. When the flow transitions to supersonic, the resulting equations become hyperbolic reflecting a change from a diffusive to a propagation dominated or advective nature. In this case the pressure at a node is a function only of the upwind conditions because pressure signals (which travel at the speed of sound) can no longer propagate upwind.

The upwind factor is used to transition the equations from diffusive to advective in nature and is based on the local fluid velocity and sound speed. When the flow is subsonic, indicated by a Mach number (v/a) less than 1, the upwind factor is 0. For supersonic flow, when the Mach number is greater than 1, the upwind factor is 1. Note that when the upwind factor equals 1, the mass flow rates at internal nodes do not cancel when the global load vector is formed. Thus a node mass flow rate must be specified; the appropriate mass flow rate corresponds to a velocity equal to the sonic velocity and the upwind fluid conditions. Thus the transition from sonic to supersonic flow (or vice versa) is part of the solution. Also, there is the possibility that the transition can occur through a shock discontinuity. The simple linear upwind elements employed in SAFSIM can capture such shocks but significant "smearing" of the shock occurs. Again, this is the problem related to the modeling of an advective process using an Eulerian mesh. More complex higher-order elements are needed for accurate shock capturing. At the system level, capturing of a shock, however smeared, is usually adequate and the smearing has little or no effect on the upwind system response. SAFSIM is not intended for the detailed study of shocks, especially in light of the inherent multidimensional nature of this phenomena.

The use of an upwind factor in the mechanical energy finite element equation allows approximate modeling of supersonic flow while retaining the accuracy of the conventional (central differenced) element for subsonic flow. An advantage of separating the mechanical and thermal energy equations is that different upwind factors can be used for the two different advective mechanisms in the fluid mechanics equations. Whereas the thermal energy equation deals with the transport of thermal energy, the mechanical energy equation deals with the transport of mechanical energy. The two equations are coupled via the equation of state and fluid properties such that their simultaneous solution provides the solution to a total energy equation. The solution of the mechanical energy equation for pressure and mass flow rate is fully implicit. Again, the solution procedure is discussed more thoroughly later in Section 6, as is the implementation of "essential" boundary conditions.

4.3 Mass Fraction Equation

The final finite element equations to be developed are those governing the mass fractions for each fluid component of a multicomponent fluid. If the fluid is comprised of a single component, the properties appearing in the thermal and mechanical energy equations are simply those of the single fluid. If the fluid is comprised of more than one component, the properties represent those of the fluid mixture. Thus the multiple components are assumed to be homogeneously mixed and in thermodynamic equilibrium, thereby negating the need for multiple momentum and energy equations. To determine the mixed fluid properties, according to the mixing rules provided in Section 3.5, it is necessary to calculate the node mass fractions for each fluid component via the conservation of mass equation. If the mixture contains N_c fluid components, only $N_c - 1$ mass fraction equations need be solved; the mass fraction for the final component is determined from Equation (3.80).

The equation governing mass conservation, in terms of the mass fraction, for fluid component *k* is developed in Section 3.4 and provided by Equation (3.81). This equation is repeated here for convenience:

$$x^k V \frac{\partial \rho}{\partial t} + x_2^k \dot{m} \chi_2 - x_1^k \dot{m} \chi_1 = S^k \tag{3.81}$$

This equation needs to be expressed for each local node of the assumed two-noded linear finite element. First, because the velocities of all fluid components are assumed to equal the velocity of the mixture (the homogeneous assumption), the mass fraction for nodes and elements is

$$\frac{\dot{m}^k}{\dot{m}} = \frac{\rho^k v^k \varepsilon A}{\rho v \varepsilon A} = \frac{\rho^k}{\rho} = x^k \tag{4.101}$$

Now, writing Equation (3.8) in terms of a single fluid component and dividing by the mixture mass flow rate allows the element mass fraction, x^k, to be expressed in terms of the node mass fractions. Thus

$$w_m x_1^k + (1 - w_m) x_2^k = x^k \tag{4.102}$$

where w_m is the mass fraction weighting factor (equal to the mass flow rate weighting factor). Because Equation (3.81) is already expressed in an integral form for a control volume, it is easy to construct the corresponding finite element equation using the direct approach. The finite element equation is

$$\left(\left[D_x^k \right] + \left[K_x^k \right] \right) \left\{ x^k \right\} = \left\{ R_x^k \right\} \tag{4.103}$$

where $[D_x^k]$ is the mass fraction density matrix for fluid component k, $[K_x^k]$ is the mass fraction advective matrix for fluid component k, $\{x^k\}$ is the node mass fraction vector for fluid component k, and $\{R_x^k\}$ is the mass fraction load vector for fluid component k. These matrices are defined for a two-noded linear finite element as

$$\left[D_x^k\right]=\frac{V}{2}\frac{\partial\rho}{\partial t}\begin{bmatrix}(1-w_m) & w_m\\ w_m & (1-w_m)\end{bmatrix}+\eta_x\frac{V}{2}\frac{\partial\rho}{\partial t}\begin{bmatrix}-(1-w_m) & -w_m\\ w_m & (1-w_m)\end{bmatrix} \tag{4.104}$$

$$\left[K_x^k\right]=\frac{\dot{m}}{2}\begin{bmatrix}-\chi_1 & \chi_2\\ -\chi_1 & \chi_2\end{bmatrix}+\eta_x\frac{\dot{m}}{2}\begin{bmatrix}\chi_1 & -\chi_2\\ -\chi_1 & \chi_2\end{bmatrix} \tag{4.105}$$

$$\left\{x^k\right\}=\begin{Bmatrix}x_1^k\\ x_2^k\end{Bmatrix} \tag{4.106}$$

$$\left\{R_x^k\right\}=\frac{S^k}{2}\begin{Bmatrix}1\\ 1\end{Bmatrix}+\eta_x\frac{S^k}{2}\begin{Bmatrix}-1\\ 1\end{Bmatrix} \tag{4.107}$$

where η_x is the upwind factor for the fluid mass fr[illegible]quation. Because no mass diffusion term is included in the original g[illegible]g mass conservation equation, the equation is purely advective in nature and only an upwind formulation is needed. A conventional element (central differencing equivalent) capability is included here for generality in case mass diffusion is added at a later time. For advection only, $\eta_x = 1$ and Equations (4.104), (4.105), and (4.107) simplify to

$$\left[D_x^k\right]=\frac{V}{2}\frac{\partial\rho}{\partial t}\begin{bmatrix}0 & 0\\ 2w_m & 2(1-w_m)\end{bmatrix} \tag{4.108}$$

$$\left[K_x^k\right]=\frac{\dot{m}}{2}\begin{bmatrix}0 & 0\\ -2\chi_1 & 2\chi_2\end{bmatrix} \tag{4.109}$$

$$\left\{R_x^k\right\}=\frac{S^k}{2}\begin{Bmatrix}0\\ 2\end{Bmatrix} \tag{4.110}$$

It is instructional to carry out the operations of Equation (4.103) for a single element, using these matrices, to demonstrate that it reduces to the governing equation [Equation (3.81)]. Keep in mind that these equations are for positive flow defined from local node 1 to local node 2. If mass flow rate and density are assumed known, the mass fraction finite element equations are static (no time-derivative terms). The dynamic nature of the equation arises from the assumed-for-now dynamic density term which is part of the fluid mechanics iterative solution procedure to be discussed in Section 6.1.

Solution of the mass fraction equation provides the mass fractions at the nodes. Because fluid properties for the mixture are evaluated for the finite elements, it is necessary to use Equation (4.102) to evaluate the element mass fraction in terms of the node mass fractions. Currently, a mass fraction weighting factor value of 1/2 is employed in SAFSIM which implies that mass sources and mass accumulation are spatially uniform within the element.

5.0 Special-Purpose Finite Elements

The finite element equations for fluid mechanics described in the previous sections are referred to as standard finite elements. To provide greater modeling versatility, three special-purpose finite elements are included in SAFSIM: (1) Choked Flow Boundary Finite Element, (2) Compressor/Pump Finite Element, and (3) Distributed Flow Manifold Finite Element. A fourth special-purpose element is planned to allow simulation of a turbine. Using these special-purpose finite elements invokes particular algorithms, which are described in the next three subsections.

5.1 Choked Flow Boundary Finite Element

With the inclusion of upwind elements in the mechanical energy finite element equations, it is possible to calculate sonic (choked) and supersonic flow with no special treatment of the equations. An alternative for accommodating choked flow at an exit is to implement the choked flow boundary finite element.✦ The choked flow equations used in SAFSIM are based on steady, adiabatic, frictionless flow of an ideal gas, neglecting potential energy and fluid conduction, and can be found in most fluid mechanics textbooks such as Reference 2. These equations and their SAFSIM implementation are described here.

Consider the flow exiting a pipe that may contain an orifice or converging nozzle as depicted in Figure 5. The flow exits into an infinite plenum with a pressure, p_b, specified by a boundary condition. Choked flow exists when the flow at location 2 reaches sonic velocity because pressure conditions at the exit can no longer propagate upstream. (Irreversible expansion occurs from p_2 to p_b.) In terms of pressure, choked flow exists when

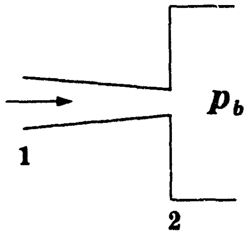


Figure 5. Choked Flow Schematic

$$\frac{p_b}{p_0} < \frac{p^*}{p_0} = \left(\frac{2}{\gamma+1}\right)^{\frac{\gamma}{\gamma-1}} \tag{5.1}$$

where p_0 is the stagnation pressure (the pressure that would exist if the fluid were brought to rest adiabatically), p^* is the critical pressure (the pressure corresponding to a Mach number of unity), and γ is the constant-pressure specific heat divided by the constant-volume specific heat. The stagnation pres-

✦ As mentioned earlier, upwind elements for the mechanical energy equation are not fully implemented in the current version of SAFSIM. Thus use of the choked flow boundary element provides a simple but valuable choked flow modeling capability.

sure can be determined in terms of the local fluid conditions by the following equation:

$$p_0 = p\left[1 + \frac{\gamma - 1}{2} M^2\right]^{\frac{\gamma}{\gamma - 1}} \tag{5.2}$$

where the square of the Mach number, M^2, can be expressed in terms of the mass flow rate as

$$M^2 = \frac{\dot{m}^2}{\rho^2 A^2 a^2} = \left(\frac{\dot{m}^2 RT}{p^2 A^2 \gamma}\right) \tag{5.3}$$

where the ideal gas law [Equation (3.83)] for density and Equation (3.102) for the sonic velocity of an ideal gas are applied. Note that Equation (5.1) is derived from Equation (5.2) with $M = 1$. Equation (5.2) can be expressed in terms of fluid conditions at location 1 and again at location 2, with the two resulting equations combined to provide an expression for p_1 in terms of p_2. Thus

$$p_1 = p_2\left[\frac{1 + \frac{\gamma - 1}{2} M_2^2}{1 + \frac{\gamma - 1}{2} M_1^2}\right]^{\frac{\gamma}{\gamma - 1}} \tag{5.4}$$

The first step in the procedure for calculating the flow conditions in a choked flow boundary element is to use Equations (5.1), (5.2), and (5.3) to determine if the flow is choked at location 2. The stagnation pressure is calculated using the current fluid conditions at location 1 and the fluid temperature at location 2. This is consistent with the iterative approach used to solve the mechanical energy equation in which the mass flow rate is assumed known in the determination of the pressure drop. The iterative approach is described in Section 6. (From here on, Locations 1 and 2 are associated with local nodes 1 and 2, respectively, of a choked flow boundary finite element.)

If the flow is not choked, $p_2 = p_b$ and Equation (5.4) can be solved for p_1. Because M_1 is a function of p_1, a fixed-point iterative technique is employed. Also note that γ is not known at nodes 1 and 2 because the specific heats are evaluated for the finite elements and not the nodes. Typically γ is a very weak function of temperature and pressure and this approximation is not significant. A shorter element length can be used to minimize the error if desired.

If the flow is choked, the Mach number at node 2 equals 1 but p_1 is not a function of p_b. Therefore, another equation is needed in conjunction with

Equation (5.4) to determine p_1. This equation is developed starting from the mass flow rate evaluated at node 2. Thus

$$\dot{m} = c_d \rho_2 A_2 a_2 = c_d \rho_2 A_2 \sqrt{\gamma R T_2} \tag{5.5}$$

where c_d is a discharge coefficient that can be used to account for nonisentropic flow effects. Discharge coefficients can be found in most fluid mechanics handbooks. Making use of the ideal gas law again, Equation (5.5) can be expressed in terms of p_2 to yield

$$p_2 = \frac{\dot{m}}{c_d A_2}\sqrt{\frac{T_2 R}{\gamma}} \tag{5.6}$$

Combining Equations (5.4) and (5.6) then produces

$$p_1 = \frac{\dot{m}}{c_d A_2}\sqrt{\frac{T_2 R}{\gamma}}\left[\frac{\left(1+\frac{\gamma-1}{2}\right)}{\left(1+\frac{\gamma-1}{2}M_1^2\right)}\right]^{\frac{\gamma}{\gamma-1}} \tag{5.7}$$

Again, because M_1 is a function of p_1, this equation is solved iteratively.

The next step, for both subsonic and choked flow, is to calculate an effective fluidity for the element in terms of the node pressures and element mass flow rate. This is accomplished using the mechanical energy equation given by Equation (3.47); imposing the choked flow boundary element assumptions of steady ($\chi_1 = \chi_2 = 1$), frictionless flow with negligible potential energy provides

$$\frac{1}{F} = \Re = \frac{1}{\dot{m}}\left[\frac{p_2}{\rho_2} - \frac{p_1}{\rho_1}\right] \quad \text{or} \quad F = \dot{m}\left[\frac{p_2}{\rho_2} - \frac{p_1}{\rho_1}\right]^{-1} \tag{5.8}$$

This effective fluidity is then simply used in the mechanical energy finite element matrices just as if it were a standard finite element except the time-derivative, potential energy, and fluid conduction terms are omitted. When the flow is choked, the value of p_2, originally provided as a boundary condition, is replaced with the value calculated by Equation (5.6). However, the original pressure value is retained for subsequent evaluation of the choked flow condition via Equation (5.1).

The final step is to determine an appropriate value for the mechanical power converted to thermal power, q_{flo}, which is defined by Equation (3.72) in Subsection 3.3. This value is needed in the compressible-flow thermal energy equation. Although Equation (3.72) can be used to approximate q_{flo} (with f and K equal to 0) for the choked flow finite element, an expression consistent with the steady, adiabatic, frictionless flow assumptions is developed starting with the following relationships between T_1 and T_2:

$$T_2 = T_1 \left[\frac{1 + \frac{\gamma - 1}{2} M_1^2}{1 + \frac{\gamma - 1}{2} M_2^2} \right] \tag{5.9}$$

based on choked flow theory, and

$$q_{flo} = \dot{m} c_p (T_2 - T_1) \tag{5.10}$$

neglecting fluid conduction. These two equations can be combined to give

$$q_{flo} = \dot{m} c_p T_1 \left[\frac{\frac{\gamma - 1}{2} \left(M_1^2 - M_2^2 \right)}{1 + \frac{\gamma - 1}{2} M_2^2} \right] \tag{5.11}$$

For choked flow, M_2 equals unity; otherwise, M_2 is determined using Equation (5.3) evaluated at node 2 fluid conditions. Using the q_{flo} value calculated by Equation (5.11) in the thermal energy equation provides the appropriate element exit fluid temperature, T_2, in terms of T_1, consistent with the choked flow boundary finite element formulation.

The equations for the choked flow finite element are formulated based on a pressure boundary condition at node 2 (the element exit); this is the most common boundary condition. However, because SAFSIM allows the analyst to specify either pressure or mass flow rate boundary conditions at any node, it is necessary to also formulate the choked flow equations based on a mass flow rate boundary condition at node 2.

The first step now is to determine the maximum-possible value for the mass flow rate at node 2. This can be determined by writing Equation (5.5) in terms of p^* using the ideal gas law. Thus

$$\dot{m}^* = c_d A_2 p^* \sqrt{\frac{\gamma}{RT_2}} \tag{5.12}$$

where $\dot{m}^*$ is the critical mass flow rate and the critical pressure, p^*, is determined by combining Equation (5.1) with Equation (5.2) evaluated at node 1 fluid conditions to provide

$$p^* = p_1 \left(M_1^2\right)^{\frac{\gamma}{\gamma-1}} \tag{5.13}$$

If the specified mass flow rate at the boundary is less than the critical mass flow rate, the flow is not choked and p_2 can be determined in terms of p_1 using Equation (5.4) together with Equations (5.3) and (5.9). These equations must be solved iteratively with respect to p_2 and T_2.

If the specified mass flow rate at the boundary is greater than the critical mass flow rate, the flow is choked and the boundary condition mass flow rate is replaced with the critical mass flow. The boundary condition mass flow rate is retained in case the flow subsequently becomes unchoked. For choked flow, p_2 equals p^*, which is calculated using Equation (5.13). Again, Equations (5.8) and (5.11) are used to determine values of F and q_{flo}, respectively.

As mentioned, the choked flow boundary finite element equations are based on steady, frictionless flow, neglecting potential energy and fluid conduction. These conditions are not overly restrictive for sonic flow of a gas because the time-derivative, friction, potential energy, and conduction terms are usually small compared to the kinetic energy term. Also, SAFSIM allows the inclusion of a discharge coefficient to account for any losses associated with nonisentropic flow, thereby improving simulation capability. Such discharge coefficients for orifices and nozzles are available in many fluid mechanics handbooks.

5.2 Compressor/Pump Finite Element

Because many flow systems of interest include a compressor or a pump, a special-purpose finite element is included in SAFSIM to allow the simulation of such a device. The term "compressor" is commonly used when the device is applied to a gas while the term "pump" is used when the working fluid is a liquid. For brevity, the term pump is used hereafter. Also, only a centrifugal-type pump capability is currently available, although an electromagnetic pump option for liquid metal applications is planned.

The input of mechanical work to a pump is used to increase the pressure of the fluid passing through the pump and thereby provides a driving potential for flow. In a system level program such as SAFSIM, the details of the fluid flow within the pump are not of interest. What is of interest is the overall pump flow performance as a function of the system pressure response. Such a global response can be characterized with what is referred to as a pump performance or characteristic curve. Such curves are based on steady-state experimental data and provide a relationship between pump pressure rise and delivered flow.

The dependent variable in a characteristic curve is the net pump head, H_p, which is defined as the pump pressure rise (Δp) divided by $g\rho$, where g is the acceleration due to gravity and ρ is the fluid density. This definition arises from the Bernoulli equation with changes in velocity and elevation neglected. Thus the net head is the difference between the pressure head developed by the pump and any frictional losses within the pump. The independent variable is the pump volumetric flow rate, $\dot{w}$. A plot of net head versus the volumetric flow rate, for a reference or rated pump speed, produces the pump characteristic curve.

As with the choked flow element, a relationship for an effective fluidity in terms of the node pressures and element mass flow rate is needed for the pump element. Again, the relationship is provided by the mechanical energy finite element equation for steady flow given by Equation (5.8). To facilitate transformation of the pump characteristic curve into an effective fluidity, the net head versus volumetric flow curve must be converted to a pressure rise (Δp) versus a mass flow rate curve. (Pressure rise and mass flow rate are the quantities of direct calculational interest in SAFSIM.) The conversions are easily accomplished by multiplying the net head by $g\rho$, and by multiplying the volumetric flow rate by ρ. The appropriate density multiplier is generally a rated density (the density of the working fluid used in the creation of the characteristic curve), but any value can be used because the fluidity is a function only of the slope of the characteristic curve. The appropriate g multiplier is the rated value that is almost always the average value corresponding to sea level, namely 9.8 m/s^2. The resulting curve is referred to as the modified pump characteristic curve.

As mentioned, a characteristic curve is based on a rated rotational shaft speed. The curve can be modified for different speeds using the pump similarity rules which arise from a pump dimensional analysis. These rules, in terms of rated conditions, are

$$\frac{\dot{m}}{\dot{m}_r} = \frac{s}{s_r} \tag{5.14}$$

$$\frac{\Delta p}{\Delta p_r} = \left(\frac{s}{s_r}\right)^2 \tag{5.15}$$

and

$$\frac{P}{P_r} = \frac{\rho}{\rho_r}\left(\frac{s}{s_r}\right)^3 \tag{5.16}$$

where $\dot{m}_r$ is the rated mass flow rate, s is the shaft speed in revolutions per unit time (revolutions/s in SAFSIM), s_r is the rated speed, P is the power supplied to the shaft, P_r is the rated power, and ρ_r is the rated density. The last rule for power is not needed in determining an effective fluidity for the pump but is included for completeness. The use of these similarity ruıes allows a single characteristic curve to be used for all rotational speeds.

The first step in calculating the effective fluidity is to determine the pressure rise based on the current value of mass flow rate using the modified pump characteristic curve. Next, using the current value of the pump inlet pressure, the pump outlet pressure is simply:

$$p_2 = p_1 + \Delta p \tag{5.17}$$

Use of the current values of mass flow rate and inlet pressure is consistent with the iterative solution approach used in the solution of the ﬂuid mechanics equations. The effective fluidity for the pump finite element can now be determined using Equation (5.8). However, an adjustment is made to this equation that greatly improves convergence of the iterative solution method of the mechanical energy equation. The adjusted equation is

$$F = \dot{m}\left[\frac{\chi_2}{\rho_2}p_2 - \frac{\chi_1}{\rho_1}p_1 - \underbrace{\frac{\Delta\hat{p}}{\rho}\left(\frac{s}{s_r}\right)^2}_{\text{adjustmont term}}\right]^{-1} \tag{5.18}$$

where $\Delta\hat{p}$ is the peak or maximum value of pressure rise from the modified pump characteristic curve. The mass flow rate fractions are retained for dynamic simulations. The term involving the peak pressure rise is the pressure rise adjustment term. This equation for fluidity is used in the solution of the node pressures. The element mass flow rate is then determined from

$$\dot{m}^{n+1} = \frac{\left\{\frac{\chi_2}{\rho_2}p_2 - \frac{\chi_i}{\rho_1}p_1 - \frac{\Delta\hat{p}}{\rho}\left(\frac{s}{s_r}\right)^2\right\}^{n+1} - \frac{L}{\rho^{n+1}\varepsilon A\Delta t_f}\dot{m}^n}{\Re^{n+1} - \frac{L}{2\rho^{n+1}\varepsilon A\Delta t_f}\left(1+\frac{\rho^n}{\rho^{n+1}}\right)} \tag{5.19}$$

This equation is just Equation (4.95) modified for the pump element to include the pressure rise adjustment term. Also, the time-derivative terms are retained for dynamic simulations but the potential energy term is omitted because it is already accounted for in the resistance (inverse fluidity) via the pump characteristic curve. Including the adjustment term can be thought of as a transformation of variables. Inclusion of this term in Equation (5.19) is therefore an inverse transformation. Although the pump characteristic curves are based on steady-state data, inclusion of the dynamic mass flow rate and density terms in the mechanical energy equations allows approximation of the dynamic response of the fluid. For steady-state calculations, the dynamic terms simply disappear.

The need for adjustment of the fluidity for a pump element arises because the introduction of the pump effective fluidity into the assembled global fluidity matrix results in a matrix that is not positive definite. The problem with nonpositive definite matrices is that many of the numerical methods for solving matrix equations will not always work. Mathematically, a positive definite matrix is one for which the product $\lfloor X \rfloor [Y] \{X\}$ is greater than zero for all nonzero vectors $\{X\}$. This criteria is not very informative to an engineer and at the risk of offending mathematical purists, an engineering explanation is offered.

When the fluidity is negative, flow in the positive direction (from node 1 to node 2) produces a pressure drop. If the fluidity is positive, as occurs with a pump finite element, a positive flow produces a pressure rise.[✦] This is possible because power is being supplied to the fluid by the pump. An element with a large area increase (an expansion) can also produce a positive fluidity. A global fluidity matrix with positive fluidity components can result in a nonpositive definite matrix. The iterative approach used in SAFSIM for solution of pressure and mass flow rate does not always converge if the global fluidity matrix is not positive definite. The adjustment to the fluidity for the pump finite element greatly enhances mass flow rate convergence.

The final step in the formulation of the pump finite element is to develop an equation for q_{fl_o}, which is defined by Equation (3.72) in Subsection 3.3. As with the choked flow element, Equation (3.72) can be used to approximate q_{fl_o}. For pumps with a liquid working fluid, in which density variations are small, the approximation is very good. For gases, although Equation (3.72) still provides reasonable results, an improvement can be made by returning to Equation (3.26) and analytically evaluating the pressure drop integral (term K4 of the mechanical energy equation). This integral can be analytically evaluated if a relationship between pressure and density is known. Assuming an isentropic

[✦] In structural mechanics, nonpositive definite matrices result when a node displaces in a direction opposite to the direction of the applied force.

process for an ideal gas provides the following relationship from basic thermodynamics:

$$\frac{p}{\rho^{\gamma}} = \text{constant} \tag{5.20}$$

The integral now becomes

$$\text{term K4} = \int_{p_1}^{p_2} v\varepsilon A dp = \dot{m} \int_{p_1}^{p_2} \frac{1}{\rho} dp = \dot{m} \int_{p_1}^{p_2} p^{-\left(\frac{1}{\gamma}\right)} dp = \dot{m} \frac{p_1}{\rho_1} \frac{\gamma}{\gamma - 1} \left[\left(\frac{p_2}{p_1} \right)^{\frac{\gamma-1}{\gamma}} - 1 \right] \tag{5.21}$$

Equation (3.72) can now be expressed for a pump with an ideal gas working fluid as

$$q_{flo} = \dot{m} \frac{p_1}{\rho_1} \frac{\gamma}{\gamma - 1} \left[\left(\frac{p_2}{p_1} \right)^{\frac{\gamma-1}{\gamma}} - 1 \right] \tag{5.22}$$

The friction terms are omitted because their effect is already accounted for in the pump characteristic curve. This equation provides the value of q_{flo} that is required for solution of the compressible thermal energy equation. Compressors that provide large pressure rises can produce significant increases in fluid temperature associated with gas compression. The analytic expression of Equation (5.22) results in accurate predictions of this temperature increase and allows the use of a single finite element for the pump.

If the pump is not energized (no mechanical power is being supplied and the shaft is not rotating), the pump finite element is treated as a standard finite element. In this case, the input values of Δz, $(L/D)_e$, and K along with a calculated wall friction factor are used to determine the pressure drop and mass flow rate. SAFSIM allows the analyst to define a cutoff rotational speed below which the pump is assumed to not be energized. Appropriate values of added loss coefficients can be used to simulate a locked rotor in the forward or reversed flow direction for a nonenergized pump. Although fluid inertial effects are accounted for in the pump finite element, the inertia associated with the pump's rotating components must be incorporated via the input specification for the pump speed. A user-supplied function can be incorporated to include this inertia; however, a special general-purpose function is planned for a later version of SAFSIM to provide this capability.

5.3 Distributed Flow Manifold Finite Element

Modeling the flow in a manifold is required in many flow systems of interest. In a manifold, two or more flow streams merge into a common plenum. Two typical manifold configurations are shown schematically in Figure 6. The first configuration consists of multiple channels connected to a single larger channel (discrete feed) and the second configuration consists of a plenum fed by a continuous source of flow along its entire length (continuous feed). The first configuration is applicable to tees. The second configuration arises when many channels feed into the plenum. The channels may be pipes or the interstitial spaces of a porous media. Also shown accompanying the two configurations are diagrams of associated 1-D finite element network representations. The dots represent nodes connecting the elements.

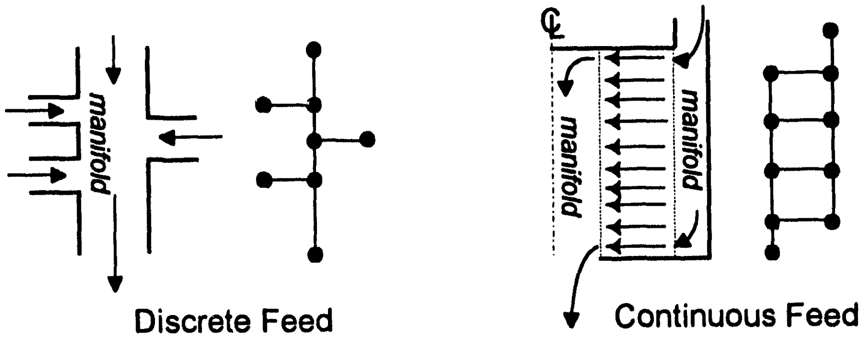


Figure 6. Distributed Flow Manifold Configurations

The vertical channels in Figure 6 are referred to as manifolds and the horizontal channels are referred to as feed lines. If the manifold volume is large or if the manifold flow rate is much greater than that of the feed lines, the manifold acts as an infinite plenum with respect to momentum exchange and no special considerations are required in the finite element model. In fact, multiple connections at a single node are allowed in SAFSIM to simulate multiple coincident connections to a plenum. However, if the feed lines and manifold are of similar diameter or if their flow rates are comparable, the influx of momentum from the feed lines to the manifold (or vice versa in the case of reverse flow) can significantly effect the manifold pressure response. In such instances, the turning and merging of the flow streams is not adequately modeled with a simple 1-D approximation.

Figure 7 is included to better illustrate the need for a special-purpose finite element to model manifolds. This figure shows schematically a single large-diameter feed line connected to the manifold. Adjacent to the schematic is a corresponding finite element model with the elements numbered for later reference. Also included on this figure are plots showing the fluid velocity profiles, as a function of position, that would be predicted using standard and distributed flow manifold elements.

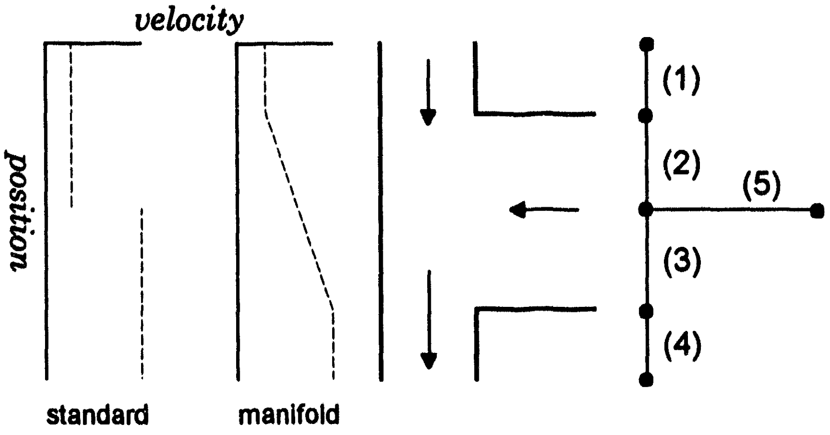


Figure 7. Velocity Profiles for Standard and Manifold Finite Elements

With the standard 1-D finite element, there is a step increase of the fluid velocity in the manifold at the connecting node to account for the inflowing fluid from the feed line. Though the velocities upstream and downstream of the node are different, the velocity profiles in the individual finite elements are uniform (assuming no heat addition or area change). Thus there is no pressure change due to velocity changes (the velocity pressure✦) within each element. This is the best you can do with a simple 1-D model. In actuality, as the flow enters the manifold from the feed line, the fluid velocity in the manifold continuously increases. Though the flow in a channel can be approximated as 1-D, the merging of flow streams is an inherently multidimensional event.

The manifold element provides a pseudo two-dimensional capability and simulates flow entering (or exiting) along the element's entire length and not just at the nodes. In other words, the feed line inflow is "distributed" along the length of the element. The distribution is assumed to be linear for simplicity.

The governing equation for a manifold finite element is derived by modifying term K2 of the conservation of momentum equation given by Equation (3.2). This term represents the net efflux of momentum (in terms of kinetic energy) through the control surfaces. In the original derivation, this term is simplified by assuming just two ports for entering and exiting flow. For the manifold element, term K2 is written assuming three ports. Thus

✦ The phrase "dynamic pressure" is more common than "velocity pressure" when referring to the kinetic energy term in the mechanical energy equation. However, velocity pressure is used in this document and the word "dynamic" is reserved for reference to time-deriviative terms.

$$\text{term K2} = \int_{A_2} \tfrac{1}{2}\varepsilon\rho v^3 dA - \int_{A_1} \tfrac{1}{2}\varepsilon\rho v^3 dA - \int_{A_3} \tfrac{1}{2}\varepsilon\rho v_3 v^2 dA \tag{5.23}$$

where A_3 is the flow area of the third port (the feed line), and v_3 is the velocity of the fluid in the feed line at the connection to the manifold. Following the procedure outlined in Subsection 3.2, the kinetic energy correction factor is introduced and Equation (5.23) is simplified to

$$\text{term K2} = \tfrac{1}{2}\alpha_2\left(\dot{m}v^2\right)_2 - \tfrac{1}{2}\alpha_1\left(\dot{m}v^2\right)_1 - \tfrac{1}{2}\alpha_3\dot{m}_3 v^2 \tag{5.24}$$

where α_3 and $\dot{m}_3$ are, respectively, the kinetic energy correction factor and mass flow rate for port 3. The kinetic energy correction factor for port 3 is not the same as described in Subsection 3.2. The correction factor accounts for the radial variation of velocity, which is different for a feed line entering a manifold than for flow in a pipe. However, the difference is not expected to be large, especially for turbulent flow. Therefore, like the standard element, α_3 is assumed to equal α (where α is the correction factor for the element) and α equals 2 for laminar flow and 1.05 fcr turbulent flow.

Now term K2 can be written as

$$\text{term K2} = \tfrac{1}{2}\alpha\left[\dot{m}_2 v_2^2 - \dot{m}_1 v_1^2\right] - \tfrac{1}{2}\alpha\dot{m}_3 v^2 \tag{5.25}$$

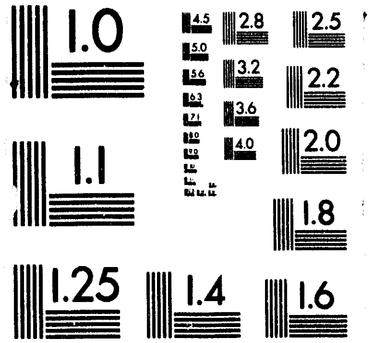
Introducing the mass flow rate fractions and expressing velocities in terms of mass flow rates, after some manipulation, produces

$$\text{term K2} = \dot{m}\frac{\alpha\dot{m}^2}{2\varepsilon^2}\left[\frac{\chi_2^3}{\rho_2^2 A_2^2} - \frac{\chi_1^3}{\rho_1^2 A_1^2} - \frac{\dot{m}_3}{\dot{m}\rho^2 A^2}\right] \tag{5.26}$$

If the mass flow rate at port 3 is zero, this expression is identical to that for the standard finite element. The new K2 term is now incorporated in the mechanical energy equation, resulting in a new equation for the flow resistance. Thus Equation (3.46) becomes

$$\Re = \frac{\dot{m}}{2}\left[\frac{\alpha}{\varepsilon^2}\left(\frac{\chi_1^3}{\rho_1^2 A_1^2} - \frac{\chi_2^3}{\rho_2^2 A_2^2} + \frac{\dot{m}_3}{\dot{m}\rho^2 A^2}\right) - \frac{1}{\rho^2}\left\{\frac{f}{A^2}\left[\frac{L}{D} + \left(\frac{L}{D}\right)_e\right] + \frac{K}{A_{\min}^2}\right\}\right] \tag{5.27}$$

This is the resistance (the reciprocal of fluidity) used in the finite element equations for a distributed flow manifold finite element.


1.0
4.5
5.0
5.6
6.3
7.1
8.0
9.0
2.8
3.2
3.6
4.0
2.5
2.2
2.0
1.1
1.8
1.25
1.4
1.6

The next step is to modify the conservation of mass equation to include the third port of the manifold finite element. Equation (3.6) written for a manifold element now becomes

$$V\frac{\partial \rho}{\partial t} + \dot{m}_2 - \dot{m}_1 - \dot{m}_3 = S \tag{5.28}$$

Writing this in terms of $\dot{m}_3$ for substitution into Equation (5.27) gives

$$\dot{m}_3 = V\frac{\partial \rho}{\partial t} - S + \dot{m}_2 - \dot{m}_1 \tag{5.29}$$

With this approach, the mass flow rate at port 3 can be evaluated in terms of the node 1 and node 2 mass flow rates. These node mass flow rates are determined by adding one-half of the mass flow rates from all connecting feed line elements, excluding other distributed flow elements and manifold terminator elements. A manifold terminator element is any standard or special-purpose element (as flagged by the analyst) that connects to a manifold element but does not represent a feed line.

Referring to Figure 7, elements (1) and (4) would be flagged as terminator elements, elements (2) and (3) are manifold elements, and element (5) is a standard element representing a feed line. Thus in SAFSIM, all elements connected to a manifold element are assumed to be feed lines unless designated as manifold or terminator elements. Again referring to Figure 7, half of the feed line flow from element (5) is apportioned to manifold element (2) and half to manifold element (3); this is why one-half of the flow is used in the determination of the node 1 and node 2 mass flow rates. Once the node 1 and node 2 mass flow rates are determined for the manifold element, the mass flow rate fractions are calculated using Equations (3.7).

The distributed flow manifold finite element allows the velocity pressure associated with the merging of flow streams in a manifold to be approximated. Because the feed line flow rates are not known in advance, an iterative approach is used, consistent with the iterative solution technique for the mechanical energy equation to be discussed in the next section.

The final step is to account for the unrecoverable losses associated with the turning of the flow. The analyst can include *K* factors for the element to account for the turning losses or the analyst can select a built-in correlation that calculates an effective friction factor based on current flow conditions. An interface is also available to allow inclusion of user-specified correlations for special problem-specific flow geometries.

Inclusion of a correlation from Reference 6, that accounts for turning losses in a tee or wye, is planned but not yet available. However, correlations for transpiration flow in cylindrical and annular geometries have been implemented. Transpiration flow effects arise for configurations similar to the continuous feed example of Figure 6 when flow passes through a porous structure. n this example, if the feed lines consist of a packed bed contained by porous structures at the inner and outer radial positions, then the geometry of the inner manifold is cylindrical and the geometry of the outer manifold is annular. When the direction of flow is from the feed lines to the inner manifold, a "blowing" transpiration flow condition exists. When the flow is in the opposite direction (from the inner manifold into the feed lines) a "sucking" condition exists.

For blowing conditions, the boundary layer at the wall is thickened resulting in a decrease in the wall friction factor (relative to flow in a channel with no transpiration flow). For sucking conditions, the boundary layer at the wall is thinned and the wall friction factor increases. Correlations[20] are available in SAFSIM for both blowing and sucking flow conditions. Also, these correlations account for turning losses in addition to transpiration losses.

For blowing conditions, the correlation is given by the following equations:

$$\frac{\bar{f}}{f} = \frac{4\varphi}{\left[1+\sqrt{1+B_M}\right]^2} \tag{5.30}$$

$$\varphi = 1 - 0.287B_0 + 0.115B_0^2 \quad \text{for } B_0 \le 2.5 \quad \text{and} \quad \varphi = 1 \quad \text{for } B_0 > 2.5 \tag{5.31}$$

$$B_0 = \left|\frac{2v_\perp}{vf}\right| \tag{5.32}$$

and

$$B_M = \left|\frac{2v_\perp}{v\bar{f}}\right| \tag{5.33}$$

where $\bar{f}$ is the effective friction factor accounting for transpiration, f is the unmodified friction factor based on flow in the absence of transpiration ($v_\perp = 0$), B_M is the blowing parameter, φ is an empirical correction factor, B_0 is the blowing parameter based on the unmodified friction factor, $v_\perp$ is the fluid velocity perpendicular to the wall at port 3 (the feed line fluid velocity), and v is the average fluid velocity in the manifold element (parallel to the wall). The feed line fluid velocity is calculated as

$$v_{\perp} = \frac{\dot{m}_3}{\rho_3 \varepsilon_3 A_3} \qquad (5.34)$$

where ρ_3, ε_3, and A_3 are the density, porosity, and flow area at port 3, respectively. This velocity is positive for blowing conditions. Because the blowing parameter in Equation (5.30) is a function of the effective friction factor, an iterative solution approach is used.

For sucking conditions, the transpiration flow correlation is given by the following equations:

$$\frac{\bar{f}}{f} = \frac{S_0}{\left[1 - \exp(-S_0)\right]} \qquad (5.35)$$

and

$$S_0 = \left| \frac{2v_{\perp}}{vf} \right| \qquad (5.36)$$

where S_0 is the sucking parameter. For sucking flow, $v_{\perp}$ is negative. Because Equation (5.35) contains an exponential, it is replaced with a more efficient curve fit given by

$$\frac{\bar{f}}{f} = 0.987838 + 0.5348S_0 + 0.0731984S_0^2 - 0.00376855S_0^3 \qquad (5.37)$$

The effective friction factor calculated by Equation (5.30) for blowing flow and Equation (5.37) for sucking flow are incorporated in the mechanical energy equation by imposing a corresponding equivalent length-to-diameter ratio that can be determined as

$$\left(\frac{L}{D}\right)_e = c_a \frac{L}{D}\left[\frac{\bar{f}}{f} - 1\right] \qquad (5.38)$$

where c_a equals 1 for cylindrical geometry and 1/2 for annular geometry. This value of $(L/D)_e$ is then used in the evaluation of the resistance given by Equation (5.27). The equivalent length-to-diameter ratio, $(L/D)_e$, is first introduced in

Equation (3.30). The value of $(L/D)_e$ for manifold elements is continuously updated to reflect changing flow conditions. The ratio $\bar{f}/f$ is the ratio of the friction factor with transpiration and turning losses included to the friction factor that would exist at the same element mass flow rate but without transpiration flow. Reference 21 provides an example of the use of distributed flow manifold elements in the simulation of a particle bed fuel element.

6.0 Fluid Mechanics Solution Procedure

The governing equations for fluid mechanics are developed in Section 3 based on integral balance equations for conservation of mass, momentum, and energy. The equations are expressed in terms of a mechanical energy equation and a thermal energy equation, along with an equation for the mass fractions of multicomponent fluids. Finite element equations are then formulated for these equations in Section 4. The finitc element equations are developed for two-noded linear 1-D elements. The next topic for discussion is the solution procedures employed in SAFSIM to solve the finite element equations, along with the associated numerical methods.

Solution of the fluid mechanics equations is complicated by the requirement that four dependent variables (pressure, mass flow rate, fluid temperature, and mass fraction) must be determined simultaneously. The thermal energy equation cannot be solved without knowing the mass flow rate and the mechanical energy equation cannot be solved without knowing the density. Both equations include the additional complication that both diffusive and advective processes are involved. Furthermore, the fluid properties (μ, c_p, c_v, k_f), required for both thermal and mechanical energy equations, cannot be evaluated without knowing the pressure, temperature, and mass fraction. And the mass fraction cannot be calculated without knowing the mixture mass flow rate and density.

This interdependency problem is often addressed with some type of iterative procedure. SIMPLE[22] (semi-implicit method for pressure-linked equations) and its many variations are perhaps the best known of such iterative procedures. A more general iterative procedure has been developed for SAFSIM. This procedure, RISES (relaxed iteration of sequential equation sets), uses fixed-point iteration implemented sequentially for multiple equation sets with under-relaxation or overrelaxation applied separately for each equation set. Although this procedure is not very sophisticated, it works reasonably well for a large class of static and dynamic flow problems. This is appropriate and desirable for a general-purpose flow simulator such as SAFSIM because of the almost unlimited problem definitions available to the analyst. The RISES procedure may not be the most efficient procedure for any given problem, but it is relatively robust for a diverse range of problems.

In addition to a steady-state capability, SAFSIM provides the analyst with a choice of three transient mode capabilities: (1) quasi-static -- only boundary conditions and thermal loads (such as convection heat transfer) are time dependent; (2) partially dynamic -- mass flow rate and fluid temperature time-derivative terms are included; and (3) fully dynamic -- the fluid density time-derivative term, in addition to the mass flow rate and temperature time-derivative terms, is included (this option is not fully implemented). The quasi-static transient mode is useful for many gas flow problems when the dynamic terms are negligible. The partially dynamic transient mode is useful for liquid flow problems when mass accumulation can be ignored. These options increase execution speed and add to the versatility of SAFSIM. Also, the analyst can bypass solution of the mechanical energy equation (pressures and mass flow rates are provided by the user) or include multiple fluid components. An advantage of the RISES procedure is that it can be universally applied for steady-state and all transient modes of operation for compressible or incompressible flow. The procedure also works if the mechanical energy equation is bypassed or if multiple fluids are included. Thus the iterative solution procedure is consistent with the primary SAFSIM development goal of versatility.

A basic outline of the RISES solution procedure is provided in Figure 8. There are six equation sets that are solved in sequence, namely: (1) the mechanical energy equation (if activated) with conservation of mass embedded, (2) the fluid thermal energy equation (the structure energy equation is also solved if desired and coupled to the fluid energy equation), (3) the mass fraction equation (for multicomponent fluids), (4) the equation of state (fluid density), (5) the fluid property equations, and (6) the conservation of mass equation for the mass flow rate fractions if mass source or dynamic density terms are included. The arrow from equation set 6 to equation set 1 indicates an outer iteration based on fluid density. A density relaxation parameter between 0 and 2 can be applied if desired. Also, inner iterations are employed in equation sets 1 and 2. Mass flow rate is the iteration parameter for equation set 1 (with relaxation), and element fluid temperature is the iteration parameter for equation set 2 (again with relaxation if desired).

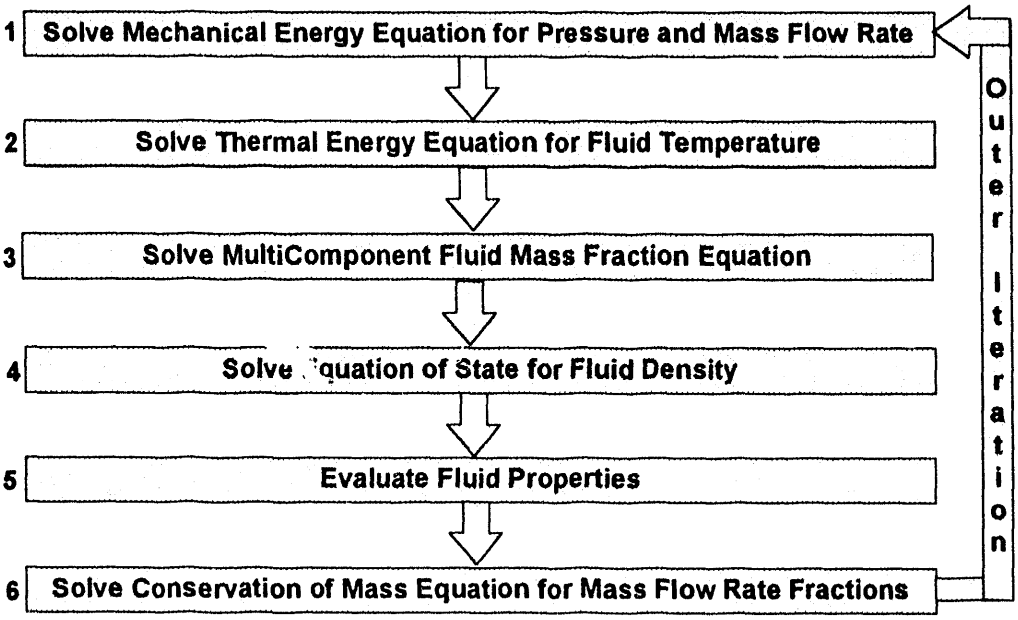

1
Solve Mechanical Energy Equation for Pressure and Mass Flow Rate
2
Solve Thermal Energy Equation for Fluid Temperature
3
Solve MultiComponent Fluid Mass Fraction Equation
4
Solve Equation of State for Fluid Density
5
Evaluate Fluid Properties
6
Solve Conservation of Mass Equation for Mass Flow Rate Fractions
Outer Iteration

Figure 8. Basic Outline of the RISES Solution Procedure

Although the RISES procedure is relatively straightforward, there are some subtleties in its implementation that enhance convergence and robustness. These subtleties are discussed as the fluid mechanics solution procedure is described in more detail in the following subsections.

6.1 Mechanical Energy Equation

Solution of the mechanical energy finite element equation set is perhaps the most difficult because of the nonlinear relationship between pressure and mass flow rate. Compressibility effects, flow regime (laminar, transition, or turbulent) effects, and factors that can produce nonpositive definite matrices, such as area expansions and pumps, also add to the difficulties.

The first step torward solution of the mechanical energy equation is the creation of a global fluidity matrix and load vector by the assembly of the finite element local fluidity matrices and load vectors. The assembly procedure for two-noded finite elements is relatively simple; however, a basic understanding of the process is important with respect to the selection of global node numbering schemes and the application of boundary conditions. Therefore, Appendix B provides an overview of the assembly procedure along with a few comments on bandwidth minimization.

The local fluidity matrices and load vectors are determined using Equation (4.96) for the fluidity matrix and Equations (4.97) through (4.100) for the load vector.✦ The mass flow rate and density values from the most recent iteration are used in this determination. The density remains constant throughout the mass flow rate inner iteration and is updated after each outer iteration following solution of the thermal energy and mass fraction equations. If a quasi-static or partially dynamic transient solution is sought, the appropriate dynamic terms in the dynamic mass flow rate load vector [Equation (4.100)] are set to zero.

After the global fluidity matrix and load vectors are assembled, essential boundary conditions must be applied. As mentioned in Section 4.2, mass flow rate at a node appears automatically in the load vector as a natural boundary condition. However, at least one node pressure boundary condition must be imposed to render the matrix equations nonsingular. The number of equations is reduced by one for every node with a specified pressure boundary condition. For the remaining equations involving a node with a specified boundary condition, the corresponding fluidity matrix term multiplied by the boundary node pressure is moved into the load vector. Pressure and mass flow rate boundary conditions can be applied at any super node in a flow network but care must be

✦ Because upwind elements for the mechanical energy equation are not fully implemented, theses equations are currently used with the upwind factor, η_F, equal to zero.

taken to ensure that the equation set is not under or overspecified. (Super nodes are defined in Subsection 6.1.3.)

If a steady-state or quasi-static solution of a closed-loop network is desired, a special closed-loop pressure boundary condition is required to render the pressure equations nonsingular. This boundary condition is also required for the partially dynamic solution because the dynamic density term is neglected in the governing equations. (In a closed loop, there may be no "boundaries" at which to specify a pressure boundary condition, unless an additional line off the closed loop is included. If there are no pressure boundaries, there are an infinite number of loop pressure distributions that can satisfy the quasi-static flow equations. This is analogous to rigid-body motion in structural mechanics; rigid-body motion is avoided by specifying at least one node displacement.) It should be noted that a closed-loop network must also include at least one convection exchange surface to render the fluid temperature equations nonsingular.

For gas-filled closed-loop networks, a special closed-loop pressure boundary condition can be applied to any of the loop interior nodes. In addition to an initial pressure guess, the network gas density times the gas constant must be provided. In a closed-loop network, the density may vary spatially around the loop but the average density for the network remains constant.

The ideal gas law is used to determine the network average pressure and based on this average pressure, a pressure-correction factor is calculated. This pressure correction factor is given as

$$\alpha_p = \frac{(\rho R)_L \overline{T}_L}{\overline{p}_L} \tag{6.1}$$

where $(\rho R)_L$ is the specified network density-gas constant product, $\overline{T}_L$ is the average fluid temperature for the network, and $\overline{p}_L$ is the average network pressure. The average fluid temperature is determined such that enthalpy is conserved globally. Thus

$$\overline{T}_L = \frac{\sum_{j=1}^{N_e} \left(\rho V c_p T\right)_j}{\sum_{j=1}^{N_e} \left(\rho V c_p\right)_j} \tag{6.2}$$

and the average network pressure is determined by volume-weighting the element pressures according to the following:

$$\overline{p}_L = \frac{\sum_{j=1}^{N_e} (pV)_j}{\sum_{j=1}^{N_e} V_j} \qquad (6.3)$$

where j is the element index number, and N_e is the number of finite elements in the network. The pressures for all of the nodes in the network are then multiplied by the pressure correction factor, which is updated every density outer iteration. Use of the correction factor thus provides a closed-loop average network pressure consistent with the ideal gas law applied to a constant fluid density network. Thus an increase in the fluid temperature results in an increase in network pressure (and vice versa) due to thermal expansion.

Two additional closed-loop pressure boundary condition options are planned for SAFSIM. One option will provide pressure control for networks using a low vapor pressure liquid, such as a liquid metal, and the other option for a high vapor pressure liquid such as water. The model for the liquid metal option will include provisions for an expansion volume filled with a cover gas. The model for the water option will include provisions for a water-vapor filled volume with heaters and coolers (a pressurizer model).

For distributed flow manifold elements (described in Subsection 5.3), the mass flow rate fractions must be evaluated before creation of the global fluidity matrix. The mass flow rate fractions are evaluated using Equations (3.7).

6.1.1 Pressure Solution

After the global fluidity matrix and load vectors are created, and boundary conditions applied, the matrix equations are solved for the node pressures. Depending on the problem, the resulting coefficient matrix may or may not be diagonally dominant or positive definite. For example, as discussed in Subsection 5.2, the inclusion of pumps and flow area expansions may lead to a nonpositive definite matrix. The inclusion of upwind elements in the mechanical energy equation also changes the properties of the matrix. Therefore, three different equation solvers are available for solution of the pressure matrix equations to provide program robustness while maintaining reasonably high execution speeds. The three solvers are (1) Gauss-Seidel (an iterative solver), (2) Cholesky decomposition (a direct solver), (3) and Gauss elimination (also a direct solver). The analyst may select any of these three solvers or let SAFSIM implement each of the three solvers sequentially until a solution is obtained.

Gauss-Seidel is in general the fastest of the three solvers because it operates only on the nonzero entries of the matrix. It also requires the smallest amount of storage space. However, if flow conditions change during a simulation such that the coefficient matrix is not diagonally dominant, convergence of this iterative method may not occur. Diagonal dominance is defined such that

$$|e_{ii}| > d_m \sum_{\substack{j=1 \\ j \neq i}}^{N_{eq}} |e_{ij}| \tag{6.4}$$

where e is a matrix entry, d_m is a factor that equals 1 for strict diagonal dominance, i is the matrix row index, j is the matrix column index, and N_{eq} is the number of equations in the matrix that equals the number of nodes minus the number of pressure boundary conditions. SAFSIM uses this equation, with d_m set to 0.97 to provide some margin, to check for diagonal dominance. If diagonal dominance is not satisfied, the Gauss-Seidel solver is skipped to save time because convergence is not guaranteed. The Gauss-Seidel solver is implemented with a user-specified relaxation parameter to speed convergence. The default value of 1.1 for the relaxation parameter works well for many problems but may not be optimal for all problems. The details of the Gauss-Seidel method with relaxation, along with the Cholesky decomposition and Gauss elimination methods, can be found in Reference 23.

If Gauss-Seidel iteration is skipped or if it does not converge, the Cholesky decomposition solver is tried next (Cholesky can be skipped if requested by the analyst). This is a direct solver (no iteration required) that operates only on entries within the semibandwidth of the coefficient matrix and is therefore relatively fast. Its shortcoming is that it does not work if the coefficient matrix is not positive definite nor symmetric. In these situations, it is skipped and Gauss elimination is employed. This is another direct solver. It uses partial pivoting to minimize round-off error and to avoid division by zero. Also, it operates only on matrix entries within the bandwidth of the coefficient matrix to greatly increase speed. Although this is usually the slowest of the three solvers, it is also the most robust and versatile. The use of three solvers in SAFSIM adds to its robustness. A fourth solver planned is known as LU decomposition with iterative refinement. This solver is similar to Gauss elimination but has been designed to correct for round-off error associated with large matrices.

6.1.2 Mass Flow Rate Solution

Recall that the fluidity (the reciprocal of resistance) is determined using the most recent value of the mass flow rate, as are the mass flow rate fractions. Therefore, the node pressures determined by solution of the pressure matrix equations are not necessarily consistent with the mass flow rates. Using the newly calculated node pressures, the element mass flow rates can be updated as part of an inner iteration for mass flow rate. For fully dynamic flow simulations, the updated mass flow rate is calculated using Equation (4.95), which is just the finite element mechanical energy equation [Equation (3.47)] after rearrangement. This equation is repeated here with a *u* subscript appended to the updated mass flow rate for clarity. Thus

$$\dot{m}_u^{n+1} = \frac{\left(\frac{\chi_2}{\rho_2} p_2 - \frac{\chi_1}{\rho_1} p_1\right)^{n+1} + g_z \Delta z - \frac{L}{\rho^{n+1} \varepsilon A \Delta t_f} \dot{m}^n}{\Re^{n+1} - \frac{L}{2\rho^{n+1} \varepsilon A \Delta t_f}\left(1 + \frac{\rho^n}{\rho^{n+1}}\right)} \tag{6.5}$$

For partially dynamic flow simulations, the equation for the updated mass flow rate can be derived again starting from Equation (3.47) but setting the density time-derivative term to zero. Thus with mass accumulation neglected, the updated mass flow rate is given by

$$\dot{m}_u^{n+1} = \frac{\left(\frac{\chi_2}{\rho_2} p_2 - \frac{\chi_1}{\rho_1} p_1\right)^{n+1} + g_z \Delta z - \frac{L}{\rho^{n+1} \varepsilon A \Delta t_f} \dot{m}^n}{\Re^{n+1} - \frac{L}{\rho^{n+1} \varepsilon A \Delta t_f}} \tag{6.6}$$

For steady-state and quasi-static transient mode calculations, both the dynamic density and mass flow rate time-derivative terms are set to zero and the updated mass flow rate simplifies to

$$\dot{m}_u^{n+1} = \frac{\left(\frac{\chi_2}{\rho_2} p_2 - \frac{\chi_1}{\rho_1} p_1\right)^{n+1} + g_z \Delta z}{\Re^{n+1}} \tag{6.7}$$

Recall from Subsection 4.1 that a fully implicit formulation is currently used for solution of the fluid mechanics equation because stable, oscillation-free solutions result, independent of time step. Another reason for choosing a fully implicit formulation is that it has a good chance of producing reasonable results for a variety of flow situations, although it may not be the best for any given situation. An implicit formulation appears most appropriate for a general-purpose flow simulator such as SAFSIM. It should be noted that although an implicit formulation is stable at any time step, accuracy remains very dependent on time step. Because it is possible to produce stable nonsense if the time step is too large, the analyst must carefully choose an appropriate time step value for each flow network. Automatic time step control for fluid mechanics is planned for a future version of SAFSIM.

Equations (6.5), (6.6), and (6.7) provide an updated mass flow rate in terms of the current iteration values of mass flow rate and resistance. Before updating the Reynolds number and returning to the pressure equations for the next mass flow rate inner iteration, a relaxation parameter is applied to the updated and previous values of mass flow rate according to the following:

$$\dot{m}_R^{n+1} = \lambda_m \left(\sqrt{\dot{m}_u^{n+1} \dot{m}^{n+1}} \right) + (1 - \lambda_m) \dot{m}^{n+1} \qquad (6.8)$$

where the R subscript indicates the relaxed value to be used for the subsequent iteration, and λ_m is the relaxation parameter for mass flow rate iterations. The mass flow rate variable without a subscript represents the value from the previous mass flow rate iteration (not from the previous time step). This type of relaxation is referred to as square root relaxation and is very effective in accelerating mass flow rate convergence. A typical value for the relaxation parameter is 1.2, which is the default value in SAFSIM. Although the square root operation is relatively time consuming, the extra computational time is more than compensated for by the time savings associated with a reduced number of mass flow rate iterations. The square root relaxation is used for steady-state and all transient modes of operation.

For partially and fully dynamic transient simulations, the mass flow rate time-derivative term is also updated after the node pressures are calculated. However, the relaxed new-time mass flow rate value is not used. Instead, more rapid convergence is achieved if the time-derivative term is calculated using the nonrelaxed mass flow rate value; thus

$$\left(\frac{\partial \dot{m}}{\partial t} \right)_u = \frac{\dot{m}_u^{n+1} - \dot{m}^n}{\Delta t_f} \qquad (6.9)$$

where $\dot{m}^n$ is the mass flow rate value at the end of the previous time step. The time-derivative term is used in Equation (4.100) to calculate the dynamic mass flow rate load vector. This updated load vector, along with the new fluidity, is used to form the pressure equations for the next mass flow rate iteration. Note that the mass flow rate time-derivative term appears as an effective load in the pressure matrix equations.

The last step before initiating the next iteration is to update the mass flow rate fractions for fully dynamic simulations using Equations (3.9). Note that the mass flow rate fractions equal 1 if the mass accumulation term (dynamic density) is neglected and no mass source terms are included.

Mass flow rate iterations continue until the user-specified convergence criteria is met or the maximum number of iterations is exceeded. The convergence criteria is based on an element mass flow rate relative error given by

$$\varepsilon_m = \left| \frac{\dot{m}_R - \dot{m}}{\dot{m}_R} \right|^{n+1} \tag{6.10}$$

Convergence of the mass flow rate iterations provides a consistent pressure and mass flow rate solution for all elements based on the current density values. The fluid density is updated after solution of the thermal energy and mass fraction equations, in preparation for a density outer iteration.

6.1.3 Super Elements

Super element capability is provided for the solution of the mechanical energy equation. This capability allows a series of finite elements to be grouped into one equivalent super element for computational efficiency. The development for a super element begins with Equation (3.47). Writing this equation in terms of the fluid resistance yields

$$(\tilde{p}_2 - \tilde{p}_1) = \Re\dot{m} - g_z\Delta z - \frac{L}{\rho\varepsilon A}\left(\frac{\partial\dot{m}}{\partial t} - \frac{\dot{m}}{2\rho}\frac{\partial\rho}{\partial t}\right) \tag{6.11}$$

where the 1 and 2 subscripts still refer to the first and second local node of each element. This equation can then be written for each finite element of a series. A series of finite elements is defined simply as elements connected end to end with no branch connections to any of the interior nodes. The individual element equations are then summed to provide the following:

$$\sum_{j=1}^{N_S}(\tilde{p}_2 - \tilde{p}_1)_j = \sum_{j=1}^{N_S}\left[\Re\dot{m} - g_z\Delta z - \frac{L}{\rho\varepsilon A}\left(\frac{\partial\dot{m}}{\partial t} - \frac{\dot{m}}{2\rho}\frac{\partial\rho}{\partial t}\right)\right]_j \tag{6.12}$$

where N_S is the number of elements in the super element. The left-hand side of this equation can be simplified if all of the element mass flow rates in the super element are assumed equal. Expanding the left-hand side and replacing the element mass flow rates with the super element mass flow rate produces the following equation:

$$\frac{1}{\dot m_{(1)}}\left(\frac{\dot m_2}{\rho_2}p_2-\frac{\dot m_1}{\rho_1}p_1\right)+\frac{1}{\dot m_{(2)}}\left(\frac{\dot m_3}{\rho_3}p_3-\frac{\dot m_2}{\rho_2}p_2\right)+\ldots+ \frac{1}{\dot m_{(N_S)}}\left(\frac{\dot m_{N_s+1}}{\rho_{N_s+1}}p_{N_s+1}-\frac{\dot m_{N_s}}{\rho_{N_s}}p_{N_s}\right)=\frac{1}{\dot m_J}\left(\frac{\dot m_{N_s+1}}{\rho_{N_s+1}}p_{N_s+1}-\frac{\dot m_1}{\rho_1}p_1\right) \qquad (6.13)$$

where a number subscript in parentheses indicates the element number, the number subscripts without parentheses now indicate <u>global</u> node numbers (assuming the nodes are numbered sequentially starting from 1 and ending at N_S + 1), and J is the super element number. Because the element mass flow rates are assumed equal, all the interior node pressure terms cancel. Only the two pressure terms at the nodes of the super element remain. It follows that the nodes of a super element are called super nodes.

The right-hand side of Equation (6.12) can be expressed as

$$\sum_{j=1}^{N_S}\left[\mathfrak{R}\dot m - g_z\Delta z - \frac{L}{\rho\varepsilon A}\left(\frac{\partial\dot m}{\partial t}-\frac{\dot m}{2\rho}\frac{\partial\rho}{\partial t}\right)\right]_j = \mathfrak{R}_J\dot m_J - g_z\Delta z_J - \frac{\partial \dot m_J}{\partial t}\left(\frac{L}{\rho\varepsilon A}\right)_J + \dot m_J\sum_{j=1}^{N_S}\left(\frac{L}{\rho\varepsilon A}\frac{1}{2\rho}\frac{\partial\rho}{\partial t}\right)_j \qquad (6.14)$$

where $\mathfrak{R}_J$ is the super element flow resistance (the sum of the element resistances), Δz_J is the super element change in elevation (the sum of the element Δz's), and $\left(\frac{L}{\rho\varepsilon A}\right)_J$ is the sum of the corresponding element terms for the super element. Based on Equations (6.13) and (6.14), Equation (6.12) becomes

$$(\tilde p_2-\tilde p_1)_J = \mathfrak{R}_J\dot m_J - g_z\Delta z_J - \frac{\partial \dot m_J}{\partial t}\left(\frac{L}{\rho\varepsilon A}\right)_J + \dot m_J\sum_{j=1}^{N_S}\left(\frac{L}{\rho\varepsilon A}\frac{1}{2\rho}\frac{\partial\rho}{\partial t}\right)_j \qquad (6.15)$$

This is the same as Equation (6.11) written in terms of a super element. Thus a series of finite elements can be represented as a single super element, greatly reducing the number of node pressure equations to be solved. After solution of the super node pressures, Equation (6.11) is used to back out the node pressures of the individual elements. The analyst should take advantage of super elements whenever possible because they substantially increase execution speed.

For static and partially dynamic calculations, Equation (6.13) is exact because then the mass accumulation terms are neglected. The equation is not exact if mass source terms are included or if the fully dynamic solution mode is

used. Elements with mass source terms should simply be specified as individual super elements with one element each. For the fully dynamic mode, the equation still provides *approximate* solutions if mass accumulation is relatively uniform throughout the super element. As a general rule, if the flow areas of a series of elements are approximately equal, super elements can be used in dynamic mode to greatly increase execution speed while providing acceptable accuracy. In this case, mass accumulation is accounted for in the super element in an average sense, based on Equation (6.15), with all of the associated elements having an averaged mass accumulation.

Super elements can also be developed for the solution of the thermal energy equation if there is no convection heat transfer to the element from a structure heat transfer element. Because the inclusion of convection is common in system analysis, the additional bookkeeping complications of thermal super elements are not included in SAFSIM. Also, solution of the thermal energy equation is generally easier than that of the mechanical energy equation, reducing the benefit of thermal super elements.

6.2 Mechanical and Thermal Energy Equation Coupling

The next step in the RISES solution procedure is to solve the thermal energy equation set. However, before this equation set can be solved and in preparation for subsequent outer iterations, several coupling terms must be evaluated based on the solution of the mechanical energy equation set. The coupling terms are (1) the effective friction factor, $\bar{f}$; (2) the flow power, i.e., the mechanical power converted to thermal power, q_{flo}; (3) the dynamic density effective power, q_ρ; (4) the grid Peclet number, Pe_L; (5) the upwind factors, η and η_F; (6) the flow direction and associated loss coefficients, $K_{1\text{-}2}$ and $K_{2\text{-}1}$; (7) the element pressure, p; and (8) the convection heat transfer coefficients, h_c, for all exchange surfaces.

The effective friction factor is first introduced in Equation (5.30) of Subsection 5.3, which deals with distributed flow manifold elements. For this type of element, the friction factor is modified to account for transpiration flow effects. This concept is extended to all finite elements such that the effective friction factor accounts for added loss coefficients and added $(L/D)_e$, in addition to wall friction. The effective friction factor for an element, which can be determined after the mass flow rate is updated in the mechanical energy solution, is defined by the following equation:

$$\bar{f} = \frac{D}{L}\left\{ f\left[\frac{L}{D} + \left(\frac{L}{D}\right)_e\right] + \frac{A^2}{A^2_{\min}} K \right\} \tag{6.16}$$

This is the friction factor printed in the output file because it reflects the contribution of multidimensional and nonstandard flow effects based on engineering factors. This effective friction factor is also used in a correlation that is based on an analogy between heat and mass transfer in the evaluation of a heat transfer coefficient.[9] This heat transfer coefficient corresponds to the friction factor correlation provided in Equation (3.36) and is but one heat transfer coefficient option the analyst can select; heat transfer coefficient evaluation is discussed in more detail later.

As discussed in Subsection 3.3, the flow power is the mechanical power converted to thermal power and represents recoverable and unrecoverable losses from compressibility and viscous dissipation. This power can be significant when the Mach number exceeds about 0.3 or if a compressor is employed. The flow power is evaluated using Equation (3.72).

The dynamic density effective power is also discussed in Subsection 3.3. This power is associated with density variations with time and is evaluated only for the fully dynamic solution mode. The dynamic density effective power is evaluated using Equation (3.74).

The grid Peclet number is used in the calculation for the thermal energy equation upwind factor. The grid Peclet number is evaluated using Equation (4.24). Recall that the grid Peclet number is based on the dispersion-enhanced fluid thermal conductivity for flow in porous media. The conductivity enhancement factor, ξ, is evaluated at the same time the fluid conductivity is evaluated (equation set 4 of the RISES procedure which is discussed in Subsection 6.5).

The upwind factor for the thermal energy equation is evaluated using Equations (4.28), based on the grid Peclet number. The upwind factor for the mechanical energy equation, though not yet implemented, is evaluated based on the Mach number as described in Subsection 4.2.

Once the mass flow rate and the flow direction are established, by solution of the mechanical energy equation, the appropriate loss coefficients are assigned according to Equations (3.31). The flow direction information is also used to determine if the node connectivity for the thermal energy equation needs to be reevaluated. As discussed in the next section, the user can select a flow direction dependent temperature boundary condition for the fluid. The specified temperature boundary condition is implemented only if flow is into the element. If flow is out of the element, a zero heat flux boundary condition is implemented instead.

The next coupling term to be evaluated is the finite element pressure, which is used for evaluation of the equation of state. The element pressure is simply the algebraic average of the node pressures, which are determined in the mechanical energy equation solution. Thus

$$p = \frac{p_1 + p_2}{2} \tag{6.17}$$

The final coupling term is the convection heat transfer coefficient. In keeping with the engineering approach adopted for SAFSIM, convection heat transfer from an exchange surface to the fluid is modeled using Newton's law of cooling, which requires a heat transfer coefficient and a wall surface temperature. The wall surface temperature comes from solution of the structure heat transfer equations (Section 7). The heat transfer coefficients are determined using empirical correlations, which are based on the fluid type, the local fluid conditions, and the flow-structure geometry. Currently, 85 correlations are available within SAFSIM. This heat transfer coefficient library includes correlations for gases, liquids, liquid metals, internal and external flow geometries, pipe flow, porous media flow, flow over tube bundles, forced flow, natural convection flow, and thermal radiation. In addition, several user-specified correlations are available to allow the analyst to easily add problem-specific and special-purpose correlations.

SAFSIM allows the analyst to specify two correlations for each convection exchange surface. One correlation is used for laminar flow and the other is used for turbulent flow. As with the friction factor, the flow regime is based on the Reynolds number which is based on bulk fluid properties. The analyst selects, for each exchange surface, the Reynolds number above which the flow is assumed to be turbulent; the Reynolds number below which the flow is laminar is then 0.6 times the upper limit Reynolds number. Linear interpolation between the two flow regimes is used for transition flow based on the same interpolation scheme used for the friction factor [Equations (3.38) and (3.39)]. The choice of correlation is specified in the structure heat transfer geometry input. Also, Equations (3.93) and (3.94) are used for those correlations requiring fluid viscosity based on the fluid temperature at the wall. A list of the available correlations is provided in the SAFSIM input manual[24] which also contains the appropriate references for each correlation and the conditions for their use.

Because more than one exchange surface can be coupled to any fluid mechanics finite element, it is necessary to sum the $h_c A_w$ and $h_c A_w T_w$ products for each surface connected to the same fluid element. These summed products appear in Equation (4.69) and (4.70) for the convection conductive matrix and load vectors, respectively.

6.3 Thermal Energy Equation

Solution of the thermal energy equation provides the fluid temperature throughout the flow network. Like the mechanical energy equation, inner iterations are required. For the thermal energy equation, the inner iterations are with respect to the element fluid temperature. Before the thermal energy equation can be solved for the node temperatures, the fluid specific heats must be known for the elements. The constant-volume specific heat appears in the capacitance matrix while the constant-pressure specific heat appears in the advection conductance matrix. In addition, two factors are a function of the element temperature and must be evaluated before the thermal energy equation can be solved. The factors are the log-mean weighting factor for convection (used in the convection conductance matrix), and the log-mean weighting factor for the dynamic temperature term (used in the capacitance matrix).

Based on the current value for the constant-pressure specific heat, the convection log-mean weighting factor is evaluated using Equation (4.54). Likewise, based on the current value for the constant-volume specific heat, the dynamic log-mean weighting factor is evaluated using Equation (4.64). The mass flow rate remains constant throughout the temperature inner iteration.

The next step is to determine the local capacitance and conductance matrices along with the associated thermal load vectors. The capacitance matrix is determined using Equation (4.66) and the conductance matrices are determined using Equations (4.67) through (4.69). Equations (4.70) and (4.71) are used to determine the thermal load vectors. Once the local matrices and load vectors are determined, they are assembled to form global matrices and load vectors. The assembly procedure is similar to that used for the mechanical energy equation and is outlined in Appendix B. If a static solution is sought, the appropriate dynamic terms are omitted from the matrix equations.

Following creation of the global coefficient matrices and load vectors, the effective total conductance matrix [Equation (4.79)] and the effective total load vector [Equation (4.80)] are formed. The effective matrices and load vectors include the implicitness factor, which equals 1 (fully implicit) in the current version of SAFSIM.

The application of essential boundary conditions is next. Convection heat transfer and direct fluid heating appear in the load vectors as natural boundary conditions during the finite element formulation. In addition, either fluid temperature or zero heat flux boundary conditions can be specified at a fluid super node. (Boundary conditions for the thermal energy equation are applied at super nodes to be consistent with boundary conditions for the mechanical energy equation, even though super elements are not used for the thermal energy equation. Super nodes are discussed in Subsection 6.1.3.) The zero heat flux boundary condition is just a special case of the convection heat transfer natural boundary condition applied at super nodes. This condition can be used at an exit node to invoke calculation of the exit fluid temperature. Also, a zero heat flux boundary condition should be used at interior super nodes that have an applied closed-loop pressure boundary condition (see Subsection 6.1 for a description of closed-loop boundary conditions).

If a fluid temperature boundary condition is specified at a super node, the number of temperature equations is reduced by one. As mentioned in Subsection 6.2, if a temperature boundary condition is specified, it is implemented only if flow is into the element. If the flow is out of the element, the temperature boundary condition is replaced with a zero heat flux boundary condition so that the exit fluid temperature can be calculated. SAFSIM monitors the flow direction at the boundary super nodes at each iteration and adjusts the connectivity data accordingly.

After the global and effective coefficient matrices and thermal load vectors are created, and boundary conditions applied, the matrix equations are solved for the node fluid temperatures. Two equation solvers are available for solution of the temperature matrix equations: (1) Gauss-Seidel, and (2) Gauss elimination. These are the same two solvers available for solution of the pressure matrix equations. Cholesky decomposition, though available for the pressure equations for subsonic flow, is not available for the temperature equations because it is applicable only to symmetric matrices; the temperature matrix is rarely symmetric due to the advective term in the thermal energy equation.

The analyst can select either of the two solvers or let SAFSIM implement each solver sequentially to obtain a solution. Gauss-Seidel is an iterative solver and is in general the fastest of the two solvers. As with the pressure matrix equations, Gauss-Seidel iteration is bypassed if the coefficient matrix is not diagonally dominant according to Equation (6.4).

For advectively dominated flow problems, the Gauss-Seidel solver converges in only 1 or 2 iterations if the nodes are numbered sequentially in the direction of flow. Such flow direction node numbering is explained more fully in the input manual.[24] If the flow reverses direction or becomes more conductively dominated, many more iterations may be required. A relaxation parameter can be applied to this solver; the default value of 1 works well for most problems.

The Gauss elimination direct solver works best if the nodes are numbered in such a way as to minimize the bandwidth of the coefficient matrix. Bandwidth minimization is discussed in Appendix B. Gauss elimination is very robust and works whether the flow is advectively or conductively dominated, independent of flow direction. However, it is not nearly as fast as Gauss-Seidel iteration for advectively dominated flows. The inclusion of both solvers adds to the robustness of SAFSIM. A capability is planned for a future version of SAFSIM in which two sets of node numbers will be stored. One set will be flow direction dependent for use with the Gauss-Seidel solver and the other set will minimize the bandwidth for use with Gauss elimination. This will maximize computational efficiency for all possible flow situations and simplify input model creation.

After the node fluid temperatures are obtained, the element fluid temperatures are updated using Equation (4.39), which expresses the element temperature in terms of the node temperatures and the log-mean weighting factor for convection. Using this equation results in what is referred to as a log-mean element fluid temperature. Note that if no convective exchange surfaces are coupled to a fluid mechanics finite element, the log-mean weighting factor equals 1/2. This corresponds to a modified Stanton number of zero ($A_w = 0$). With a weighting factor of 1/2, the fluid temperature is just the algebraic average of the two local node fluid temperatures.

A relaxation parameter for fluid temperature is then applied to the updated fluid temperature according to

$$T_R^{n+1} = \lambda_T T_u^{n+1} + (1 - \lambda_T) T^{n+1} \tag{6.18}$$

where T_R is the relaxed element fluid temperature, λ_T is the element temperature relaxation parameter, T_u is the updated element temperature, and the temperature variable without a subscript represents the value from the previous temperature inner iteration. A temperature relaxation parameter of 1 works well for many flow problems with reasonably behaved fluid properties.

According to Reference 25, the log-mean temperature is the appropriate temperature to use for property evaluation in the evaluation of friction factor correlations for nonisothermal flow through porous media. Based on this finding, the log-mean fluid temperature is used in SAFSIM for the calculation of the fluid properties for the finite element for all flow geometries. With the calculation of the relaxed element fluid temperatures, the specific heats are updated in preparation for additional temperature inner iterations. The new specific heats are then used to update the log-mean weighting factors along with the capacitance and advection conductance matrices. Temperature iterations continue until the user-specified convergence criteria is met or the maximum number of element temperature iterations is exceeded. Similar to the mass flow rate inner iterations, the convergence criteria is based on an element temperature relative error given by

$$\varepsilon_T = \left|\frac{T_R - T}{T_R}\right|^{n+1} \tag{6.19}$$

where ε_T is the fluid temperature relative error.

6.4 Mass Fraction Equation

Third in the sequence of the RISES solution procedure is the solution of the mass fraction equation for multicomponent fluids. If only a single fluid is involved, this step is bypassed. Solution of the mass fraction equation is much simpler than solution of either the mechanical or thermal energy equations. Unlike those two equations, solution of the mass fraction equation does not require inner iterations.

As discussed in Subsection 4.3, the governing equation for mass fraction is just the conservation of mass equation written for a fluid component. Also mentioned is the fact that the finite element equations for mass fraction are static; i.e., they don't contain a mass fraction time-derivative term.

The first step in the solution of the mass fraction equation is the determination of the local dynamic density and advective matrices, along with the mass fraction load vector. Using the current values for fluid density and mass flow rate, Equation (4.108) is used to calculate the dynamic density matrix and Equation (4.109) is used to calculate the advective matrix. Equation (4.110) is used to calculate the load vector.

The local coefficient matrices and load vectors are then assembled to create a global mass fraction coefficient matrix and load vector. Boundary conditions are applied next. A mass source term for a fluid component appears in the load vector as a natural boundary condition. Also, fluid component mass fraction boundary conditions can be applied at any super node. Like the thermal energy equation, super elements are not used for the mass fraction equations, but boundary conditions are applied at super nodes for consistency with the mechanical energy equation. Also, the mass fraction boundary condition depends on the flow direction. For flow out of an element, a zero mass fraction gradient boundary condition is applied to allow calculation of the exit mass fraction for the fluid component.

After application of the boundary conditions, the mass fraction equations are solved using either Gauss-Seidel iteration or Gauss elimination. These are the same two solvers used for solution of the temperature matrix equations. Because the mass fraction equations are purely advective, Gauss-Seidel iteration converges in a single iteration provided the nodes are numbered based on flow direction. Refer to Subsection 6.3 for additional comments on use of the Gauss-Seidel solver.

After solution of the mass fraction equations for the node mass fractions, the element mass fractions are evaluated using Equation (4.102). The element mass fractions are used in the determination of the mixture fluid properties. Because no inner iterations are needed, neither a relaxation parameter nor a relative error for convergence are required for the mass fraction equation. For a fluid mixture of N_c fluid components, solution of the mass fraction equations is repeated $N_c - 1$ times. The mass fraction for the final component is determined from Equation (3.80).

6.5 Equation of State, Properties, and Conservation of Mass

Upon solution of the pressure, mass flow rate, fluid temperature, and mass fraction equations, the equation of state and fluid properties can be updated. The fluid density is evaluated for both elements and nodes using Equation (3.83) for ideal gases, Equation (3.87) for ideal liquids, or user-specified equations via the built-in equation of state interface. The fluid dynamic viscosity is calculated using either Equation (3.89), Equation (3.90), or user-specified equations. The two specific heats are evaluated using Equations (3.95) and (3.96), or user-specified equations. Fluid thermal conductivity is calculated using Equation (3.98) or user-specified equations. Finally, the conductivity enhancement factor is determined using Equation (3.101) and the sonic velocity is determined using either Equation (3.102) for gases or Equation (3.103) for liquids.

For multicomponent fluids, the mixture density and fluid properties are evaluated using the mixing rules given by Equations (3.84), (3.91), (3.97), and (3.99) for ideal gases and Equations (3.88), (3.92), (3.97), and (3.100) for ideal liquids. (Mixing rules for liquid-gas combinations or for two-phase fluids are not available, although they may be implemented in a future version of SAFSIM.)

Fluid density is used as the outer iteration control variable of the RISES solution procedure. With the updating of density based on the current values of pressure and temperature, a relaxation parameter is applied based on the following equation:

$$\rho_R^{n+1} = \lambda_\rho \rho_u^{n+1} + \left(1 - \lambda_\rho\right)\rho^{n+1} \tag{6.20}$$

where ρ_R is the relaxed element fluid density, λ_ρ is the element density relaxation parameter, ρ_u is the updated element density, and the density variable without a subscript represents the value from the previous outer iteration. A density relaxation parameter of 1 works well for most flow problems in which the density is a smooth function of temperature and pressure.

As mentioned in Subsection 4.2, conservation of mass at the nodes is incorporated in the mechanical energy finite element equations. In the finite element assembly procedure, mass flow rates at internal nodes cancel (unless a mass flow rate boundary condition is specified or the flow is choked) and the dynamic density term appears in the load vector. The time derivative of density is updated using the relaxed densities from the current and previous time steps. Thus

$$\left(\frac{\partial \rho}{\partial t}\right)_u = \frac{\rho_R^{n+1} - \rho^n}{\Delta t_f} \tag{6.21}$$

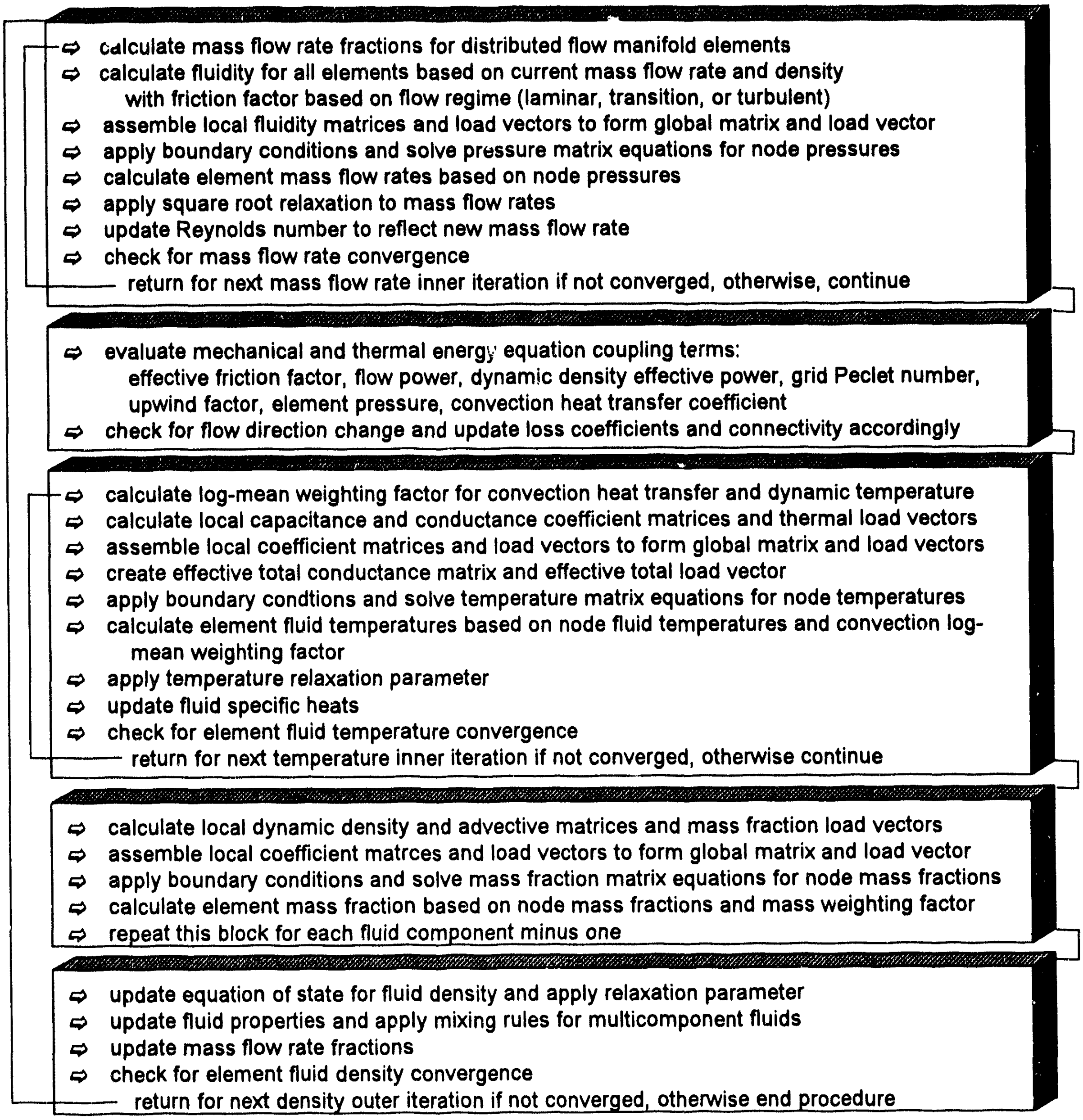
For fully dynamic calculations, the mass flow rate fractions appear in the fluidity matrix and also must be updated. The mass flow rate fractions are determined based on the latest density time-derivative value using Equations (3.9). The last step in the outer iteration is to calculate the pressure-correction factor for flow networks with a closed-loop pressure boundary condition. Equations (6.1), (6.2), and (6.3) are used for this calculation.

The convergence criteria for outer iterations is based on an element density relative error given by

$$\varepsilon_\rho = \left|\frac{\rho_R - \rho}{\rho_R}\right|^{n+1} \tag{6.22}$$

where ε_ρ is the fluid density relative error.

This completes the solution of the fluid mechanics equations using the RISES solution procedure. The interaction with and coupling to the other physics modules is described in Section 10.0, after description of the structure heat transfer and reactor dynamics modules. To supplement the basic outline of the RISES procedure provided in Figure 8, a more detailed outline is provided in Figure 9. This detailed outline summarizes the steps described in Subsections 6.1 through 6.5.



⇨ calculate mass flow rate fractions for distributed flow manifold elements
⇨ calculate fluidity for all elements based on current mass flow rate and density
with friction factor based on flow regime (laminar, transition, or turbulent)
⇨ assemble local fluidity matrices and load vectors to form global matrix and load vector
⇨ apply boundary conditions and solve pressure matrix equations for node pressures
⇨ calculate element mass flow rates based on node pressures
⇨ apply square root relaxation to mass flow rates
⇨ update Reynolds number to reflect new mass flow rate
⇨ check for mass flow rate convergence
return for next mass flow rate inner iteration if not converged, otherwise, continue

⇨ evaluate mechanical and thermal energy equation coupling terms:
effective friction factor, flow power, dynamic density effective power, grid Peclet number, upwind factor, element pressure, convection heat transfer coefficient
⇨ check for flow direction change and update loss coefficients and connectivity accordingly

⇨ calculate log-mean weighting factor for convection heat transfer and dynamic temperature
⇨ calculate local capacitance and conductance coefficient matrices and thermal load vectors
⇨ assemble local coefficient matrices and load vectors to form global matrix and load vectors
⇨ create effective total conductance matrix and effective total load vector
⇨ apply boundary condtions and solve temperature matrix equations for node temperatures
⇨ calculate element fluid temperatures based on node fluid temperatures and convection log-mean weighting factor
⇨ apply temperature relaxation parameter
⇨ update fluid specific heats
⇨ check for element fluid temperature convergence
return for next temperature inner iteration if not converged, otherwise continue

⇨ calculate local dynamic density and advective matrices and mass fraction load vectors
⇨ assemble local coefficient matrces and load vectors to form global matrix and load vector
⇨ apply boundary conditions and solve mass fraction matrix equations for node mass fractions
⇨ calculate element mass fraction based on node mass fractions and mass weighting factor
⇨ repeat this block for each fluid component minus one

⇨ update equation of state for fluid density and apply relaxation parameter
⇨ update fluid properties and apply mixing rules for multicomponent fluids
⇨ update mass flow rate fractions
⇨ check for element fluid density convergence
return for next density outer iteration if not converged, otherwise end procedure

Figure 9. Detailed Outline of the RISES Solution Procedure

7.0 Structure Heat Transfer

The structure heat transfer physics module allows the simulation of conduction heat transfer in any number of structures using a 1-D finite element model. The structure finite elements can be convectively or radiatively coupled to any of the fluid mechanics finite elements through the use of exchange surfaces. The exchange surfaces can also be used to conductively or radiatively couple the finite elements of one structure to those of another structure using function-controlled variables. This gives the analyst a tremendous capability for modeling complex flow systems despite the simple 1-D formulation. Even pseudo 2-D models can be constructed if desired. Also, SAFSIM is structured to allow inclusion of a full 2-D or 3-D conduction capability at a later date. A thermal radiation enclosure capability is also envisioned.

The governing equation for the structure heat transfer physics module is known as the heat conduction equation. Its solution is simpler than the solution of the fluid mechanics equations because it involves a single dependent variable (temperature), and because conduction is strictly a diffusive process. Thus the equation is not complicated by advective terms as is the case with the fluid mechanics equations. Also, the network capability for the fluid mechanics elements, which allows parallel and serial connections of elements to form complex networks, is not needed for the 1-D structure elements. Instead, the structure elements are always connected in series, resulting in a simple tridiagonal coefficient matrix.

The network capability of the fluid mechanics elements combined with the multiple exchange surface capability of the structure elements allows the analyst to construct models of complex flow systems. An example of a relatively complex system is a liquid-to-gas heat exchanger. One flow network with a liquid is used on the tube side with another network with a gas on the shell side. Exchange surfaces then couple the inside surface of the tubes to the liquid while additional exchange surfaces couple the outside surface to the gas using appropriate heat transfer coefficients for the different tube and shell side geometries. Additional exchange surfaces can be used to include thermal radiation heat transfer if desired. This generalized structure heat transfer capability enhances the primary SAFSIM development goal of versatility.

7.1 Governing Equation

As mentioned, the governing equation for the structure heat transfer physics module is known as the heat conduction equation. Its derivation is based on an energy balance for a differential volume in which the time rate of change of energy in the volume (the energy storage term) equals the net efflux of energy from the surface of the volume (in terms of heat flux) plus the energy generated in the volume. The energy storage term is expressed in terms of temperature and a volumetric heat capacity (density times specific heat), and Fourier's law of conduction is used to express the heat fluxes in terms of temperature and a diffusion constant (the thermal conductivity). Fourier's law, which is based on empirical observation, states that heat flux is proportional to the temperature gradient, with thermal conductivity as the constant of proportionality. The complete derivation of the conduction equation can be found in countless texts on heat transfer. Two such recommended texts are *Fundamentals of Heat Transfer*,[26] and *Heat Transfer*.[27] The governing equation for conduction heat transfer in a 1-D structure is

$$\rho_s c \frac{\partial T_s}{\partial t} = k_s \frac{\partial^2 T_s}{\partial x^2} + q_s''' \tag{7.1}$$

where ρ_s is the structure density, c is the structure specific heat, T_s is the structure temperature, t is time, k_s is the structure thermal conductivity, x is a coordinate position, and q_s''' is the volumetric heat source. In this form of the conduction equation, the thermal conductivity is assumed independent of position. This equation, as expected, is the same as the governing equation for the fluid thermal energy equation [Equation (3.77)] without an advective term.

7.2 Finite Element Formulation

As with the fluid mechanics equations, the finite element method is used in SAFSIM to transform the conduction equation into algebraic expressions for numerical solution. A brief overview of the finite element method is provided in Section 4.0. Like the fluid thermal energy equation, the method of weighted residuals is used in the development of the finite element equation for structure heat transfer. The method of weighted residuals requires that the residual error times a weighting function equals zero in an average sense over the solution domain. Mathematically, this is expressed as

$$\int_\Omega W_i \left[\rho_s c \frac{\partial T_s}{\partial t} - k_s \frac{\partial^2 T_s}{\partial x^2} - q_s''' \right] d\Omega = 0 \tag{7.2}$$

where Ω is the solution domain (the finite element volume), and W_i is the finite element weighting function. This integral formulation is sometimes called the weak form of the governing equation.

Because the heat conduction equation finite element formulation parallels that of the fluid thermal energy equation (provided in Subsection 4.1), most of the formulation details are not repeated here. Using linear interpolation functions, I_i [Equations (4.7)], and their corresponding gradient interpolation functions, B_i [Equations (4.11)], for the structure temperature, the individual terms of the finite element equations are expressed, for a two-noded finite element, by the following integrals:

$$[C_s] = \int_0^{L_k} \rho_s c\{W\}\lfloor I \rfloor A_k dx \tag{7.3}$$

$$[K_k] = \int_0^{L_k} k_s\{W\}\lfloor B \rfloor A_k dx \tag{7.4}$$

$$[K_c] = \int_0^{L_k} h_c\{W\}\lfloor I \rfloor P_h dx \tag{7.5}$$

$$\{R_{q'''}\} = \int_0^{L_k} q_s'''\{W\} A_k dx \tag{7.6}$$

$$\{R_{q''}\} = \int_0^{L_k} q_s''\{W\} P_h dx \tag{7.7}$$

$$\{R_c\} = \int_0^{L_k} h_c T\{W\} P_h dx \tag{7.8}$$

where $[C_s]$ is the structure element capacitance matrix, $[K_k]$ is the conduction conductance matrix, $[K_c]$ is the convection heat transfer conductance matrix, $\{R_{q'''}\}$ is the volumetric heating load vector, $\{R_{q''}\}$ is the heat flux load vector, $\{R_c\}$ is the load vector for convection heat transfer to an exchange surface or surfaces, L_k is the element conduction length, A_k is the element conduction area, h_c is the heat transfer coefficient, T is the fluid exchange temperature (the exchange temperature for the fluid thermal energy equation is the wall temperature, T_w), and P_h is the heated perimeter of the exchange surface. For multiple exchange surfaces for an element, the individual load vectors are simply added together. The heat flux load vector, $\{R_{q''}\}$, is not included in the fluid thermal energy equation but is included in the structure conduction equation. It appears as a result of the conduction surface integral provided in Equation (4.2). For structures, Equation (4.3) is modified to include a heat flux term in addition to the convection term. This allows the analyst to specify a heat flux for an exchange surface and adds modeling versatility to the structure heat transfer model.

The next step in the development of the heat transfer finite element equations is to define the weighting functions. Because no advective term appears in the finite element equations for conduction, the upwind elements used in the fluid thermal energy equation are not needed for the structure conduction equation. Thus the Bubnov-Galerkin formulation is employed in which the weighting functions are set equal to the interpolation functions. The integrals in Equations (7.3) through (7.8) are now evaluated to give

$$[C_s] = \frac{\rho_s c V_s}{6}\begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix} \tag{7.9}$$

$$[K_k] = \frac{k_s A_k}{L_k}\begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix} \tag{7.10}$$

$$[K_c] = \frac{h_c A_{ex}}{6}\begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix} \tag{7.11}$$

$$\{R_{q'''}\} = \frac{q_s''' V_s}{2}\begin{Bmatrix} 1 \\ 1 \end{Bmatrix} \tag{7.12}$$

$$\{R_{q''}\} = \frac{q_s'' A_{ex}}{2}\begin{Bmatrix} 1 \\ 1 \end{Bmatrix} \tag{7.13}$$

$$\{R_c\} = \frac{h_c A_{ex} T}{2}\begin{Bmatrix} 1 \\ 1 \end{Bmatrix} \tag{7.14}$$

where V_s is the element volume equal to $A_k L_k$, and A_{ex} is the exchange surface area equal to $P_h L_k$. The exchange area appears as the wall area (A_w) in the fluid thermal energy equation when the conduction element is convectively coupled to the fluid element.

In SAFSIM, the consistent form of the capacitance matrix, Equation (7.9), is replaced with a lumped capacitance matrix. The lumped form in which the element capacitance is evenly split between the two nodes is given by

$$[C_s] = \frac{\rho_s c V_s}{2}\begin{bmatrix}1 & 0\\0 & 1\end{bmatrix} \qquad (7.15)$$

The lumped form is derived from the consistent form by simply adding the entries for each row, assigning the sum to the diagonal, and zeroing the off-diagonal entries. Essentially, the lumped form is equivalent to a finite difference approach. In some instances, use of the lumped capacitance matrix suffers from a small loss of solution accuracy for transient simulations compared to use of the consistent matrix. However, for two-noded elements, the loss of accuracy is insignificant, especially with respect to other inherent inaccuracies such as those due to material property uncertainties. Also, the lumped approach avoids the nonphysical spatial oscillations that sometimes occur with the consistent approach if used with too coarse of a mesh. The analyst should be warned that in such instances, although the lumped matrix provides physical and intuitive results, it may be at the expense of reduced accuracy.

A schematic of a sample finite element model for structure heat transfer is provided in Figure 10. The nodes are numbered 1 through 5, and the corresponding four element numbers are in parentheses.

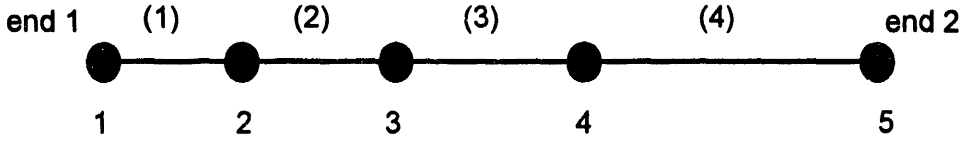


Figure 10. Structure Heat Transfer Finite Element Model Schematic

For transient simulations, the conduction equation can be solved in quasi-static or dynamic mode. In quasi-static mode, the temperature time-derivative term is set to zero and only boundary conditions and the thermal loads from convection to a fluid change with time. This mode is appropriate if the time constant for the structure is much smaller than that of the fluid. The quasi-static mode has the advantage of removing time step restrictions for the structure. In dynamic mode, the time-derivative term is included.

Following the procedure outlined in Equations (4.72) thrnugh (4.81) of the fluid mechanics thermal energy equation, the global dynamic finite element equation for structure conduction is written as

$$[\overline{K}_t]\{T_s\}^{n+1} = \{\overline{R}_t\} \qquad (7.16)$$

where $[\overline{K}_t]$ is the effective total conductance matrix defined as

$$\left[\overline{K}_t\right] = \frac{1}{\Delta t_s}\left[C_s\right] + \theta_{HT}\left[K_t\right] \tag{7.17}$$

and $\left\{\overline{R}_t\right\}$ is the effective total load vector defined as

$$\left\{\overline{R}_t\right\} = \left[\frac{1}{\Delta t_s}\left[C_s\right] - \left(1-\theta_{HT}\right)\left[K_t\right]\right]\left\{T_s\right\}^n + \left(1-\theta_{HT}\right)\left\{R_t\right\}^n + \theta_{HT}\left\{R_t\right\}^{n+1} \tag{7.18}$$

where Δt_s is the structure time step, θ_{HT} is the implicitness factor for heat transfer, and $[K_t]$ and $\{R_t\}$ are the total conductance and load vectors, respectively, given by

$$\left[K_t\right] = \left[K_k\right] + \left[K_c\right] \tag{7.19}$$

and

$$\left\{R_t\right\} = \left\{R_{q'''}\right\} + \left\{R_{q''}\right\} + \left\{R_c\right\} \tag{7.20}$$

The load vector depends on the heat transfer coefficient and the fluid temperature, which are determined as part of the fluid mechanics solution. Because of the explicit coupling between the fluid mechanics and structure heat transfer physics models, only the values at the beginning of the system level time step are available. (The explicit coupling of the physics modules is explained more fully in Section 10.) Equation (7.18) is modified to reflect this, providing the following revised equation for the effective total load vector:

$$\left\{\overline{R}_t\right\} = \left[\frac{1}{\Delta t_s}\left[C_s\right] - \left(1-\theta_{HT}\right)\left[K_t\right]\right]\left\{T_s\right\}^n + \left\{R_t\right\}^n \tag{7.21}$$

The material properties are evaluated at the beginning of each structure heat transfer sub-time step. (Each structure can have its own sub-time step that can be less than or equal to the system level time step.) Although an iterative approach could be used to determine the properties at the end of each sub-time step, such an iterative approach is not used in SAFSIM. Thus the capacitance and conductance matrices are evaluated once at the beginning of each sub-time step. In general, using the beginning-of-sub-time step property values provides sufficient accuracy for a system engineering simulation, especially in consideration of the uncertainties associated with most material properties. Smaller sub-time steps can be used if required to account for properties that are strongly dependent on temperature.

A lumped capacitance matrix [Equation 7.15)] is used in SAFSIM because in addition to simplicity, it allows easy implementation of an automatic implicitness algorithm. This algorithm automatically determines the optimum (with respect to accuracy) implicitness factor for each element of the structure, for each sub-time step. As discussed in Subsection 4.1, there are several common selections for the implicitness factor, θ_{HT}. A factor of 0 results in a fully explicit formulation, which is easy to implement but is undesirable in a general-purpose program because of the associated time step restriction required for stability [Equation (4.82)]. Implicitness factors greater than or equal to 1/2 ensure stability at any time step but provide variable levels of accuracy. Recall that a stable solution may exhibit spatial oscillations; although these oscillations diminish in time (they grow in time for explicit algorithms), they can lead to significant inaccuracies. For any given time step, there is an implicitness factor that provides optimum accuracy. Such an optimum implicitness factor is determined for each element of a structure based on a finite element adaptation of an algorithm[28] for finite difference equations. This algorithm is given by the following equation:

$$\theta_{HT} = \frac{1}{1-\exp(-\lambda_s)} - \frac{1}{\lambda_s} \tag{7.22}$$

where λ_s is a dimensionless time step defined as

$$\lambda_s = \frac{\Delta t_s}{\Delta t_E} \tag{7.23}$$

where Δt_E is the explicit time step limit calculated for a finite element as

$$\Delta t_E = \frac{\dfrac{\rho_s c V_s}{2}}{\dfrac{k_s A_k}{L_k} + \sum_{l=1}^{N_{ex}} \dfrac{2}{6}(h_c A_{ex})_l + \dfrac{1}{2}(h_c A_{ex})_{BC}} \tag{7.24}$$

where N_{ex} is the number of convection exchange surfaces assigned to the element, and the BC subscript indicates the inclusion of a convective boundary condition at a node connected to the element. This term is zero for interior elements. The explicit time step limit is the maximum time step that can be used in an explicit algorithm (θ_{HT} < 1/2) based on stability considerations. It is instructive to think of this limit as an effective time constant for the element. The larger the time constant, the slower the transient response. Based on Equation (7.22), if the time constant decreases, the optimum implicitness factor increases to a maximum value of 1 (fully implicit). If the time constant increases, the optimum factor decreases to a minimum value of 1/2 (Crank-Nicolson). The explicit time step limit for a node, which is the quantity printed in the structure heat transfer output, is determined as

$$(\Delta t_E)_i = \frac{\sum_{j=1}^{2}\left(\frac{\rho_s c V_s}{2}\right)_j}{\sum_{j=1}^{2}\left(\frac{k_s A_k}{L_k} + \sum_{l=1}^{N_{ex}} \frac{2}{6}\left[h_c A_{ex}\right]_l\right)_j + (h_c A_{ex})_{BC}} \qquad (7.25)$$

where the summation from j equals 1 to 2 indicates the sum of the terms for the two elements adjacent to node i. For nodes at a boundary, there is only one adjacent element.

To avoid evaluation of the exponential term, Equation (7.22) is replaced by the following more computationally efficient expressions:

$$\theta_{HT} = 1 - \frac{1}{\lambda_s}\left[1 - \left(1 - \frac{\lambda_s}{10}\right)^5\right] \text{ for } \lambda_s < 10 \text{ and } \theta_{HT} = 1 - \frac{1}{\lambda_s} \text{ for } \lambda_s \geq 10 \qquad (7.26)$$

The advantage of the variable implicitness factor is that it ensures stability while optimizing accuracy for a given time step. To further enhance SAFSIM's conduction heat transfer model, an adaptive time step control algorithm is included in conjunction with the automatic implicitness factor algorithm. This time step control algorithm is given by

$$\Delta t_s = f_I \Delta t_E^{\min} \qquad (7.27)$$

where f_I is a user-specified fraction greater than zero, and $\Delta t_E^{\min}$ is the minimum value of all of the node explicit time step limits for the structure. Although very simple, this algorithm, in conjunction with the automatic implicitness factor algorithm, provides efficient time step control for the structure in most situations. The default value for the fraction of the explicit limit is 1/2, which is adequate for most structures. For structures that do not strongly influence the system response, larger values of f_I can be used to reduce execution time at the expense of reduced accuracy. However, values greater than about 3 should be avoided because of possible excessive errors.

The next step is to assemble the local matrices and load vectors into a global matrix and load vector. With two-noded elements connected in series, the assembly process results in a global coefficient matrix that is tridiagonal. Thus only the diagonal, subdiagonal, and superdiagonal entries of the matrix need be stored. Further advantage of the tridiagonal matrix is realized in SAFSIM by using a tridiagonal matrix equation solver. SAFSIM uses the Thomas algorithm, which is a popular tridiagonal algorithm that can be found in most numerical methods texts such as Reference 23.

Before the conduction matrix equations are solved for the structure node temperatures, the boundary conditions must be imposed. Three types of boundary conditions can be imposed at the end nodes of a structure: (1) temperature, (2) specified heat flux, and (3) convection. Combinations of specified heat flux and convection boundary conditions can also be imposed if desired. (A special hea^t transfer coefficient can be selected to model thermal radiation heat transfer boundary conditions via the convection boundary condition option.) At least one temperature or one convection boundary condition, or at least one convection exchange surface must be specified for each structure to render the matrix equations nonsingular.

A very simple way to introduce a temperature boundary condition at an end node is to set the diagonal entry equal to 1, the offdiagonal entries equal to 0, and the corresponding load vector entry equal to the desired temperature. A heat flux boundary condition requires nothing more than including the specified heat flux in the load vector via Equation (7.13). A positive value of heat flux indicates heat leaving the structure. A zero heat flux boundary condition at an end node dictates a zero temperature gradient at that location. A convection heat transfer boundary condition is imposed by adding h_cA_{ex} to the diagonal entry, and $h_cA_{ex}T$ to the corresponding load vector entry. In addition to the boundary conditions, multiple heat flux and convection exchange surfaces can be specified for any element as part of the finite element equations.

After solution of the matrix equations, the element temperatures are determined as a function of the node temperatures using

$$T_{s,j} = \frac{T_{s,1} + T_{s,2}}{2} \tag{7.28}$$

where the j appearing in the temperature subscript indicates the element number, and the 1 and 2 appearing in the temperature subscripts indicate the first and second local nodes of the element.

Structure material properties, c and k_s, can be specified as a function of element temperature if desired. Although the governing equation for conduction is based on a spatially uniform conductivity for a differential volume, the material properties can vary from element to element dependent on the element temperature. Five options are available in SAFSIM for specifying material properties: (1) table lookup based on linear interpolation of randomly spaced temperature data, (2) table lookup based on linear interpolation of uniformly spaced temperature data, (3) an n^th^ order polynomial function of temperature (where n is user specified), (4) a power-law function of temperature, and (5) a constant.

More than one exchange surface can be specified for a conduction finite element. Because the solution is 1-D, all exchange surface temperatures are assigned the associated element temperature. Likewise, although more than one convection boundary condition can be applied to an end node, all are assigned the associated node temperature. Convection heat transfer fluxes, as printed in the output, are determined based on the following equation:

$$q_c'' = (1-\theta_{HT})[q_c'']^n + \theta_{HT}[q_c'']^{n+1} \tag{7.29}$$

where q_c'' is the convection heat transfer flux based on Newton's law of cooling:

$$q_c'' = h_c\left(T_{s,j} - T\right) \tag{7.30}$$

where T is the fluid temperature corresponding to the coupled exchange surface. According to Equation (7.29), the printed heat flux is a weighted average of the beginning- and end-of-time step values.

The conduction equation derivation is based on conduction in a solid structure. However, it also applies to conduction in a porous media as long as the specified density for the element times the element volume equals the mass of the solid in the element, and the volumetric heat source times the element volume equals the appropriate power for the element. Its important to ensure that a consistent combination of microscopic and macroscopic values of properties and dimensions is used. Also, conductivity for the porous media must be specified as an effective bed conductivity. Correlations for such effective bed conductivities are abundant in the porous media technical literature. One such correlation[29] is

$$k_{PM} = \frac{1-\varepsilon}{\left(\dfrac{1}{k_S} + \dfrac{1}{k_r}\right)} + \varepsilon k_r \tag{7.31}$$

where k_{PM} is the effective conductivity for a porous media, ε is the porosity, k_S is the conductivity of the solid, and k_r is an effective thermal radiation conductivity given by

$$k_r = 4\sigma_B e_r D T_s^3 \tag{7.32}$$

where σ_B is the Stefan-Boltzmann constant, e_r is the surface emissivity for radiation, D is the particle diameter, and T_s is the structure temperature. Equation (7.31) accounts for particle-to-particle thermal radiation and conduction at the contact points. It is important to recognize that any correlation for effective conductivity used in SAFSIM should not include the effects of fluid conduction because it is already accounted for in the fluid thermal energy equation.

As discussed in Subsection 6.2, heat transfer coefficients for each convection exchange surface and boundary condition are specified as part of the structure heat transfer input. One correlation for laminar flow and another correlation for turbulent flow must be specified. The correlations are evaluated as part of the fluid mechanics solution. The analyst must also supply a Reynolds number above which the flow is assumed to be turbulent (the Reynolds number below which the flow is assumed to be laminar is then 0.6 times this upper limit) and an equivalent diameter for heat transfer. These quantities can be different than specified for the coupled fluid mechanics finite element to allow consistency with the selected heat transfer coefficient correlation. The default value for the transition Reynolds number is that of the fluid network to which the exchange surface is coupled. The equivalent diameter for heat transfer is used in the selected correlation and in the evaluation of the Reynolds number for determination of flow regime. The default value is that of the coupled fluid mechanics element.

An example of the use of different equivalent diameters for fluid mechanics and heat transfer involves the modeling of a thermocouple heat transfer structure within the fluid flowing through a pipe. The equivalent diameter for the fluid mechanics finite element is the pipe diameter (used for friction factor determination) whereas the equivalent diameter for the structure exchange surface is some effective thermocouple diameter. Thus with respect to the pipe, this is an internal flow geometry but with respect to the thermocouple it is an external flow geometry.

An extensive heat transfer coefficient library is available in SAFSIM and includes gas, liquid, and liquid-metal correlations. Both internal and external flow geometry correlations are provided including flow in or over porous media, pipes, tube banks, annuli, spheres, and flat plates. The correlations also cover forced, free, natural, and mixed flow conditions. Also, a correlation is available to simulate thermal radiation to the fluid. Lastly, an interface is provided to allow the inclusion of special-purpose user-specified correlations. The correlations, along with appropriate references and conditions for their use, are provided in Appendix C of Reference 24. Selected heat transfer coefficient correlations are provided in Appendix C of this document.

Another SAFSIM feature that should be mentioned is the COPIES input variable. The analyst can set COPIES to any number greater than or equal to zero. This variable multiplies the exchange surface area specified in the structure input to obtain the wall exchange area used in the fluid mechanics solution. Thus

$$A_w = \text{COPIES} \cdot A_{ex} \tag{7.33}$$

where A_w is the wall exchange surface area used in the fluid mechanics solution and A_{ex} is the exchange surface area specified for the structure.

An example of the use of the COPIES variable involves the modeling of a heat exchanger with multiple tubes. If a group of tubes are expected to behave similarly, conduction heat transfer in only one tube need be modeled. COPIES is then set equal to the number of similar tubes so that the correct amount of heat is transferred to the fluid. Also, the COPIES variable can be set to zero to model supplemental structures. For example, a group of power-producing particles can be modeled as a single "lumped" structure for computational efficiency. The power of all of the particles would be assigned to the elements of the structure and an effective conductivity would be used for the elements. A single particle could then be modeled in detail to allow prediction of its internal temperature profile. To avoid double accounting of the power associated with that particle, COPIES is set to zero.

Because the structure heat transfer physics module in SAFSIM goes beyond what is normally associated with a 1-D model, the model is referred to as an enhanced 1-D model. The multiple exchange surfaces and boundary conditions, along with the COPIES variable, provide features that allow the engineering simulation of complex problems at the system level.

7.3 Automatic 1-D Finite Element Generation

To simplify construction of a structure heat transfer input model, an automatic finite element generator is provided. This element generator determines the appropriate conduction area and volume for each element of the structure based on the selected geometry. The automatic element generator accommodates four basic geometries: (1) cylindrical, (2) spherical, (3) rectangular, and (4) linear.

The input for the element generator depends on the geometry. For rectangular geometry, the width and height of the structure, along with the position of the two structure ends, are required. The conduction area for each element is simply the product of the width and height. The volume is the area times the element length. If desired, the generator calculates the element lengths as the difference between the end positions divided by the specified number of elements.

For linear geometry, the areas and positions of the two structure ends must be provided. The element conduction areas are then determined using linear interpolation of the end areas as a function of the element center position. Again, the element lengths can be specified or automatically determined as the difference between the end positions divided by the specified number of elements.

The conduction areas for the cylindrical and spherical geometry options are somewhat more involved because the governing equation for conduction [Equation (7.1)] is based on a rectangular (Cartesian) coordinate system. If, for example, conduction in a cylindrical structure is to be modeled, the rectangular finite element equation can be used with the conduction area for each element set equal to the algebraic average of the two node areas. Reasonable solutions can be achieved with this average area provided a sufficient number of elements are used. A better choice for the conduction area, however, is the log-mean area because less elements are required for a desired accuracy. For a spherical structure, the geometric mean is the appropriate choice for the conduction area. The derivations for the log-mean and geometric mean conduction areas follow.

First, the heat flow through a surface, q, is expressed as

$$q = k_s A_k \frac{dT_s}{dx} \tag{7.34}$$

where A_k is the appropriate conduction area to be determined. For an axi-symmetric cylinder of height H, the conduction area at radial position r is

$$A_k = 2\pi r H \tag{7.35}$$

where π is the circumference of a circle divided by its diameter (3.14159...), and r is the radial position. Next, Equations (7.34) and (7.35) are combined and rearranged to give

$$q \frac{dr}{r} = 2\pi H k_s dT_s \tag{7.36}$$

Both sides of this equation are then integrated from radial position r_1 to position r_2 to provide the following new expression for heat flow:

$$q = \frac{2\pi H k_s \Delta T_s}{\ln\left(\frac{r_2}{r_1}\right)} \tag{7.37}$$

where ΔT_s is the temperature at position 2 minus the temperature at position 1.

Likewise, Equation (7.34) is rearranged and integrated to yield

$$q = k_s A_k \frac{\Delta T_s}{(r_2 - r_1)} \tag{7.38}$$

The conduction area is assumed independent of position for this integration, consistent with a rectangular geometry.

Equations (7.37) and (7.38) are then combined with A_k expressed in terms of the node conduction areas. Thus

$$A_k = \frac{2\pi H(r_2 - r_1)}{\ln\left(\frac{r_2}{r_1}\right)} = \frac{A_{k2} - A_{k1}}{\ln\left(\frac{A_{k2}}{A_{k1}}\right)} \tag{7.39}$$

where the 1 and 2 subscripts indicate radial positions 1 and 2, respectively. Therefore, A_{k1} and A_{k2} are the conduction areas for the two nodes of the element. The conduction area provided by Equation (7.39) is the appropriate conduction area for a finite element in cylindrical geometry and is known as the log-mean area.

A similar procedure is used to derive the appropriate element conduction area for a spherical geometry. For a complete sphere, the conduction area at radial position r is

$$A_k = 4\pi r^2 \tag{7.40}$$

Combining this equation with Equation (7.34) and rearranging gives

$$q \frac{dr}{r^2} = 4\pi k_s dT_s \tag{7.41}$$

Both sides of this equation are then integrated from radial position r_1 to position r_2 to provide the following new expression for heat flow in a sphere:

$$q = \frac{4\pi k_s \Delta T_s}{\dfrac{1}{r_1} - \dfrac{1}{r_2}} \tag{7.42}$$

Equations (7.42) and (7.38) are combined and rearranged to provide the following expression for the element conduction area in terms of the node conduction areas:

$$A_k = 4\pi r_1 r_2 = \sqrt{A_{k1} A_{k2}} \tag{7.43}$$

This conduction area is the appropriate area for a finite element in spherical geometry and is known as the geometric-mean area.

For solid cylinders and spheres, Equations (7.39) and (7.43), respectively, are not applicable for the inner element. For the inner element, the conduction area is determined as the element volume divided by the element length; thus

$$A_k = \frac{V_s}{L_k} = \frac{V_s}{r_2} \tag{7.44}$$

The log-mean and geometric-mean conduction areas provided by Equations (7.39) and (7.43) are derived assuming complete cylinders and spheres. SAFSIM also allows automatic element generation for partial cylinders and spheres. The derivations are extended by replacing the equations for the conduction areas at a radial position [Equations (7.35) and (7.40)] with appropriate expressions for area as a function of radial position and angle.

For a cylinder, the area is

$$A_k = \int_0^\theta rH d\theta = \theta r H \quad \text{for } 0 < \theta \leq 2\pi \tag{7.45}$$

For a sphere, the area is

$$A_k = \int_0^\theta \int_0^\phi r^2 \sin\phi \, d\phi d\theta = r^2 \theta (1 - \cos\phi) \quad \text{for } 0 < \theta \leq 2\pi \text{ and } 0 < \phi \leq \pi \tag{7.46}$$

where θ is the meridonal angle, and ϕ is the azimuthal angle, both in radians. Figure 11 shows the geometry for the two coordinate systems. These equations are used in place of Equations (7.35) and (7.40) for the node areas. The node areas are used in the evaluation of the log-mean and geometric-mean conduction areas for an element based on Equations (7.39) and (7.43).

Both cylindrical and spherical geometry options for the automatic element generator require input of the inner and outer radial positions of the structure. In addition, the cylindrical option requires input of the height, *H*, and meridonal angle, *θ*. The spherical option requires input of the meridonal angle, *θ*, and the azimuthal angle, *ϕ*. The finite element volumes are determined for cylindrical geometry using

$$V_s = \int_{r_1}^{r_2}\int_{0}^{\theta}\int_{0}^{H} r\,dH\,d\theta\,d\phi = \frac{H}{2}\theta\left(r_2^2 - r_1^2\right) \tag{7.47}$$

and for spherical geometry using

$$V_s = \int_{r_1}^{r_2}\int_{0}^{\theta}\int_{0}^{\phi} r^2 \sin\phi\,d\phi\,d\theta\,dr = \theta(1-\cos\phi)\frac{\left(r_2^3 - r_1^3\right)}{3} \tag{7.48}$$

As with the other geometry options, the element lengths can be specified or automatically determined as the difference between the end positions divided by the specified number of elements.

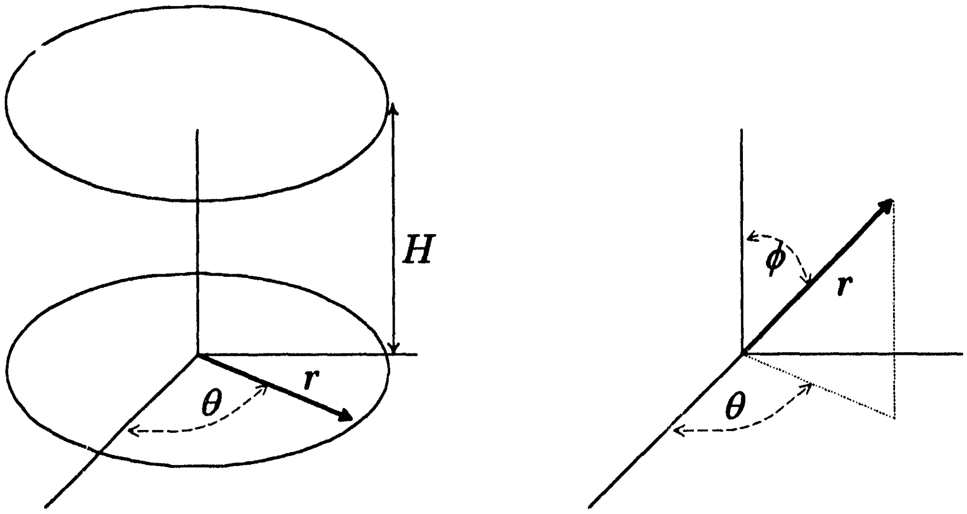


Figure 11. Geometry for Cylindrical and Spherical Coordinate Systems

8.0 Reactor Dynamics

The reactor dynamics✦ physics module allows the simulation of the time-dependent neutron density in a nuclear reactor. This is the simplest of SAFSIM's three physics modules because the numerical model is zero dimensional. Such a space-independent approach is referred to as a "lumped" or "point" model and is common in the nuclear industry because it is adequate for a broad range of problems, especially at the engineering level. In a point model, the neutron density may be a function of position, but the shape of the function is assumed to remain constant in time. Referring to the model as space-independent is somewhat misleading because, in fact, several space dependent features of the reactor are accounted for in the governing equations via engineering factors.

Any number of delayed neutron groups can be included in the model to account for the different neutron generation rates of the different isotopes produced during operation. Also, any number of decay heat groups can be included to allow prediction of the heat arising from the decay of the reactor isotopes. An arbitrary source of neutrons also can be included if desired.

The reactor dynamics physics module allows the analyst to account for reactivity feedback between the reactor neutron density (or power) and the fluid mechanics and structure heat transfer physics modules. As many feedback variables as desired can be included to account for interactions with such parameters as fluid density and structure temperature. Furthermore, the form of the feedback coefficients used to define the feedback relation is user specified. Reactor control laws also can be implemented if desired. Thus all the complex and synergistic interactions occurring within a nuclear reactor system can be simulated at the engineering level.

8.1 Governing Equations

The equations governing the dynamics of neutron density in a nuclear reactor can be derived from the time-dependent neutron diffusion equation or the transport equation. Another simpler approach is based on a neutron balance for the reactor. The details of the different approaches can be found in Reference 31. The neutron balance approach is presented here for simplicity.

The neutron balance approach, expressed qualitatively, balances the time rate of change of neutrons with the rate of neutron production minus the rate of loss. For a "point" model, the number of neutrons in the reactor is regarded as

✦ As suggested by Ott,[30] the phrase "reactor dynamics" (as opposed to "reactor kinetics") is used to indicate the inclusion of feedback effects in the governing equations.

an integral or volume-averaged quantity. The neutron balance is expressed by the following equation:

$$\underbrace{\frac{dn}{dt}}_{\substack{\text{time rate}\\ \text{of change}\\ \text{of neutrons}}} = \underbrace{(1-\gamma\beta)k_e\frac{n}{l}}_{\substack{\text{prompt neutron}\\ \text{production rate}}} + \underbrace{\sum_{i=1}^{N_n}\lambda_i C_i}_{\substack{\text{delayed neutron}\\ \text{production rate}}} + \underbrace{q_n}_{\substack{\text{neutron}\\ \text{source}}} - \underbrace{\frac{n}{l}}_{\substack{\text{neutron}\\ \text{loss rate}}} \tag{8.1}$$

where n is the neutron density (or any quantity proportional to the number of neutrons in the reactor), t is time, γ is an energy correction factor that accounts for the fact that delayed neutrons are created at lower energies than prompt neutrons and therefore experience different (usually lower) leakage and capture rates, β is the total delayed neutron fraction, which is the fraction of neutrons that are delayed (i.e., not produced promptly), k_e is the effective neutron multiplication factor, which is the number of neutrons produced (prompt or delayed) per neutron lost for a finite reactor of a given geometry (i.e., corrected for leakage), l is the effective neutron lifetime (also corrected for leakage), i is the group index number, N_n is the number of delayed neutron precursor groups, λ_i is the delayed neutron decay constant for precursor group i, C_i is the delayed neutron precursor concentration for group i, and q_n is the extraneous neutron source.

Based on these definitions, n/l can be interpreted as the neutron loss rate; therefore, $k_e n/l$ is the neutron production rate. Multiplying this by the fraction of neutrons produced promptly $(1 - \gamma\beta)$ gives the prompt neutron production rate. The delayed neutron precursor decay constant is the probability that the precursor isotope decays and produces a delayed neutron. Thus the product of the decay constant and the precursor concentration is the rate of delayed neutron production.

Two new definitions are now introduced: the neutron generation time is defined as the effective neutron lifetime divided by the effective neutron multiplication factor, and the reactivity is defined as the change in k_e from the critical condition (where $k_e = 1$ defines a critical reactor in which the production and loss rates of neutrons are exactly balanced) divided by k_e. Thus

$$l_g = \frac{l}{k_e} \tag{8.2}$$

and

$$\rho = \frac{k_e - 1}{k_e} \tag{8.3}$$

where l_g is the neutron generation time, and ρ is the reactivity (The dash through the ρ variable is used to distinguish this variable from fluid density).

These definitions are introduced into Equation (8.1), which after rearrangement produces

$$\frac{dn}{dt} = \frac{(\rho - \gamma\beta)}{l_g} n + \sum_{i=1}^{N_n} \lambda_i C_i + q_n \tag{8.4}$$

In addition to this equation, an equation is required to predict the delayed neutron precursor concentration response for each group. The delayed neutron precursor concentration equation is again just a balance between production and creation; thus for each precursor group

$$\underbrace{\frac{dC_i}{dt}}_{\substack{\text{time rate}\\ \text{of change of}\\ \text{precursors}}} = \underbrace{\frac{\gamma_i \beta_i k_e}{l}}_{\substack{\text{precursor}\\ \text{production}\\ \text{rate}}} n - \underbrace{\lambda_i C_i}_{\substack{\text{precursor}\\ \text{loss rate}}} \tag{8.5}$$

where γ_i is the group i energy correction factor, and β_i is the group i delayed neutron precursor fraction[✦] such that

$$\gamma\beta = \overline{\beta} = \sum_{i=1}^{N_n} \gamma_i \beta_i \tag{8.6}$$

where the quantity $\gamma\beta$ is the effective total delayed neutron fraction, which is commonly assigned the variable $\overline{\beta}$. The total delayed neutron fraction for a thermal-spectrum reactor is typically about 0.007, which means that less than 1% of the neutrons are delayed. These delayed neutrons, however, have a significant influence on the control of the reactor.

Combining Equations (8.2) and (8.5) gives the following equation for delayed neutron precursor concentration as a function of time for group i:

$$\frac{dC_i}{dt} = \frac{\gamma_i \beta_i}{l_g} n - \lambda_i C_i \tag{8.7}$$

[✦] Values for λ_i and β_i can be found in Reference 32. In general, six groups are adequate to represent all the different isotopes produced during operation although additional groups may be required to account for photoneutrons associated with reactors containing beryllium.

The solution of Equations (8.4) and (8.7) provides the neutron density and delayed neutron precursor concentrations as a function of time. It should be pointed out that although n is defined as the neutron density, it also can be defined as any quantity proportional to neutron density (such as power density, power, and neutron flux) as long as consistent units are used for C_i (same as n) and q_n (same as dn/dt). From here on, n is defined as the reactor total neutron power (or the instantaneous neutron power) because this is a quantity of direct interest in a system level simulation.

In addition to Equations (8.4) and (8.7), an equation is included to predict the decay heat precursor concentrations as a function of time. This equation, which balances the production and loss of decay heat precursors, for each decay heat group j is

$$\frac{dH_j}{dt} = \beta_j^H n - \lambda_j^H H_j \tag{8.8}$$

where H_j is the group j decay heat precursor concentration, β_j^H is the group j decay heat fraction, which is the fraction of the neutron power that produces decay heat precursors for group j, and λ_j^H is the decay constant for decay heat precursor group j.✦ This equation is dependent on n but Equations (8.4) and (8.7) are not dependent on H_j. Therefore, Equation (8.8) can be solved after solution of Equations (8.4) and (8.7).

The decay heat precursor concentration is then used to predict the decay heat power using the following equation:

$$n_h = \sum_{j=1}^{N_h} \lambda_j^H H_j \tag{8.9}$$

where n_h is the decay heat power, j is the decay heat group index number, and N_h is the number of decay heat precursor groups. The prompt neutron power (also referred to as the fission thermal power) is defined as

$$n_p = \left(1 - \beta^H\right) n \tag{8.10}$$

✦ Values of λ_j^H and β_j^H can be found in Reference 33. In general, eleven groups provide an acceptable approximation of decay heat for thermal-spectrum reactors.

where n_p is the prompt neutron power, and β^H is the total decay heat fraction given by

$$\beta^H = \sum_{j=1}^{N_h} \beta_j^H \tag{8.11}$$

The total decay heat fraction, β^H, for a thermal-spectrum reactor is typically about 0.07, which indicates that at steady state about 7% of the thermal power is produced by decay heat.

The effective reactor thermal power is then the sum of prompt power and decay heat power:

$$n_e = n_p + n_h \tag{8.12}$$

where n_e is the effective thermal power.

There are four variables that define reactor power; they are repeated here to emphasize and clarify their different meanings: the total or instantaneous neutron power (n), the decay heat power (n_h), the prompt neutron or fission thermal power (n_p), and the effective thermal power (n_e). All four power variables are provided as reactor dynamics output and are available for use in the other physics modules via function-controlled variables. It should be noted that the total neutron power is the power that would be measured by a neutron detecting device whereas the effective thermal power is the power that would be transferred to a coolant and reflected in the coolant temperature increase. The two are equal only at equilibrium (steady-state) conditions.

The availability of the prompt (n_p) and delayed (n_h) components of thermal power allow the analyst to apportion the power to different locations within the reactor. For example, fission power is deposited and generated at essentially the same location (within the fuel) whereas much of the decay power, which is transmitted primarily by long-range gamma rays, is deposited far from where it is generated. This can be important in transients when the relative contributions from fission and decay power are continually changing. In addition to power apportionment based on whether the power is prompt or delayed, the analyst can spatially apportion the thermal power among various structure finite elements to account for axial, radial, and azimuthal power profiles in the reactor if desired.

The initial values of the neutron power, the delayed neutron and decay heat precursor concentrations, and the reactivity must be provided before the governing equations can be integrated (solved). If the steady-state values are desired for the initial values, they are determined in terms of the initial neutron power by setting the time-derivative terms of Equations (8.4), (8.7), and (8.8) equal to zero and rearranging. Thus

$$\rho^0 = \gamma\beta - l_g \left(\frac{\sum_{i=1}^{N_n} \lambda_i C_i^0 + q_n^0}{n^0} \right) \tag{8.13}$$

$$C_i^0 = \frac{\gamma_i \beta_i}{l_g \lambda_i} n^0 \tag{8.14}$$

and

$$H_j^0 = \frac{\beta_j^H}{\lambda_j^H} n^0 \tag{8.15}$$

where the 0 superscript indicates the initial value.

The key to coupling the reactor dynamics solution to the fluid mechanics and structure heat transfer solutions is the total reactivity, ρ. Two kinds of reactivity can be specified: programmed and feedback. Programmed reactivity represents the effect of some type of control mechanism such as control rods or drums. This allows the simulation of reactor control laws whereby the control mechanism differential reactivity worth is a function of other system parameters. Several built-in control laws are available in SAFSIM to specify the programmed reactivity as a function of time. Also, an interface is available to allow the analyst to supply their own control laws via a subroutine. With this user-specified subroutine, the analyst can define the programmed reactivity as a function of any system parameter or parameters and can include inertial effects of the control mechanism if desired.

Feedback reactivity represents the inherent effect of changing system parameters, such as reactor temperature, on the neutron population. The feedback reactivities are inherent to the system, which means they are the result of direct physical processes. An example feedback reactivity is that due to changes in fuel temperature. A change in fuel temperature changes the effective absorption rate of neutrons and thereby influences the production of neutrons. The feedback reactivities are represented by the following equation:

$$d\rho_k = \alpha_k dx_k \tag{8.16}$$

where $d\rho_k$ is the differential feedback reactivity for feedback term k, α_k is the feedback coefficient for term k, and dx_k is the term k differential of some variable x, where x may represent temperature, density, power, or any other system parameter. The total differential feedback reactivity is then

$$d\rho = \sum_{k=1}^{N_f} d\rho_k = \sum_{k=1}^{N_f} \alpha_k dx_k \tag{8.17}$$

where k is the feedback term index number, and N_f is the number of differential feedback terms.

The analyst has complete control over the number of terms and the functional form of the feedback coefficients via function-controlled variables (described in the next section). For example, a temperature coefficient may be specified as a constant or as a function of an inverse temperature. Also, the temperature can be some weighted average of several structure finite elements. A special mathematical function is provided in SAFSIM to calculate this average temperature and is presented in Appendix D. Both the functional form of the feedback reactivity and the number of function arguments can be controlled by the analyst. This control affords the analyst versatility in the simulation of complex systems.

8.2 Solution Procedure

The first step in the reactor dynamics solution procedure is to update the total reactivity to reflect changes in the programmed and feedback reactivities. The following equation is used to perform this update:

$$\rho^{m+1} = \rho^m + \left(\frac{d\rho_p}{dt}\right)\Delta t_r + \sum_{k=1}^{N_f} \alpha_k \Delta x_k \tag{8.18}$$

where ρ is the total reactivity, the m+1 superscript refers to the end of time step value and the m superscript refers to the beginning of time step value (m is used as the time level index instead of the usual n to avoid confusion with the reactor power variable), $\left(\frac{d\rho_p}{dt}\right)$ is the time derivative of programmed reactivity, Δt_r is the sub-time step for reactor dynamics, Δx_k is the change in the feedback parameter x_k, and the terms within the summation are the differential feedback reactivity terms. The programmed reactivity is evaluated every reactor dynamics sub-time step but the differential feedback reactivities are only evaluated every system level time step. Tighter coupling between the feedback terms and the other physics modules can be accomplished by decreasing the system time step if necessary.

The next step is to solve Equations (8.4) and (8.7) for the neutron power and delayed neutron precursor concentrations. The solution of such rate equations is sometimes referred to as integrating the equations. Between Equations (8.4) and (8.7), there are a total of N_n + 1 equations that must be integrated. Two integrators (described in detail in Reference 23) are available in SAFSIM: Runge-Kutta-Fehlberg (RKF) and Euler. Both integrators employ adaptive time stepping to obtain a specified relative truncation error. Also, both integrators solve the N_n + 1 equations in line (sequentially) starting with Equation (8.4), which means that current values of n and C_i are used in the solution for the future values. Iterations are not taken to converge the dependent variables to the end of time step values. The in-line approach is computationally efficient and in general is satisfactory for most problems, especially when the uncertainties introduced by other assumptions in the model are considered.

The RKF integrator is fifth-order accurate and in general is very robust. The relative truncation error for adaptive time stepping is based on the maximum difference between fourth- and fifth-order solutions for each of the dependent variables. Thus

$$\varepsilon_r = \max\left[\varepsilon_n, \varepsilon_{C_i}\right] \text{ for } i = 1 \text{ to } N_n \tag{8.19}$$

where

$$\varepsilon_n = \left| \frac{n_{5th} - n_{4th}}{n_{5th}} \right|^{m+1} \tag{8.20}$$

and

$$\varepsilon_{C_i} = \left| \frac{C_{5th} - C_{4th}}{C_{5th}} \right|_i^{m+1} \tag{8.21}$$

where ε_r is the maximum relative truncation error for reactor dynamics, ε_n is the relative truncation error for n, ε_{C_i} is the relative truncation error for the delayed neutron precursor concentration for group i, and the subscripts $5th$ and $4th$ indicate the fifth-and fourth-order solutions, respectively.

The Euler solver is offered as a very simple integrator. Although less computations are required per time step, more time steps are required to obtain the same accuracy as the RKF integrator. However, for some problems not requiring a high degree of accuracy, the Euler integrator may be more efficient. Adaptive time stepping is based on an approach known as step doubling. In this approach, a solution is obtained for each dependent variable using the current time step. Next, the time step is halved and the solution repeated. The relative truncation error for each dependent variable (n or C_i) is then approximated as the difference between the two solutions divided by the one-half time step solution. The relative truncation error is the maximum relative error for all the dependent variables. Thus

$$\varepsilon_r = \max\left[\varepsilon_n, \varepsilon_{C_i}\right] \text{ for } i = 1 \text{ to } N_n \tag{8.22}$$

where

$$\varepsilon_n = \left| \frac{n_{1/2} - n_1}{n_{1/2}} \right|^{m+1} \tag{8.23}$$

and

$$\varepsilon_{C_i} = \left| \frac{C_{1/2} - C_1}{C_{1/2}} \right|_i^{m+1} \tag{8.24}$$

where ε_r is the maximum relative truncation error for reactor dynamics, ε_n is the relative truncation error for n, ε_{C_i} is the relative truncation error for the delayed neutron group i concentration, and the subscripts 1/2 and 1 indicate the one-half time step and full time step solutions, respectively.

A third integrator using the Kagonove[34] method is planned for inclusion in SAFSIM. This integrator was developed specifically for the reactor dynamics equations and has the advantage of allowing large time steps.

The next step in the reactor dynamics solution procedure is the solution of the decay heat equations, given in Equation (8.8), for H_j as a function of the time-averaged value for n. Rather than solving these equations numerically, an analytical solution is employed given by the following equation:

$$H_j = \frac{\beta_j^H}{\lambda_j^H}\bar{n} + \left(H_j^0 - \frac{\beta_j^H}{\lambda_j^H}\bar{n} \right) \exp\left[\lambda_j^H\left(t^0 - t\right)\right] \tag{8.25}$$

where $\overline{n}$ is the time-averaged neutron power for the current time step, and again the 0 superscript indicates the initial value. The initial value for each new time step is the value from the previous time step. The time-averaged neutron power is determined as

$$\overline{n} = \frac{n^{m+1} + n^{m}}{2} \tag{8.26}$$

After solution for the H_j's, the decay, prompt, and effective thermal powers are evaluated using Equations (8.9), (8.10), and (8.12), respectively.

In preparation for the next reactor dynamics sub-time step, the sub-time step is adjusted based on the relative truncation errors determined by Equations (8.19) or (8.22). The adjustment is made using the following algorithm:

$$\Delta t_r^{\text{new}} = \Delta t_r \left(\frac{\varepsilon_r^d}{\varepsilon_r} \right)^{\frac{1}{5}} \tag{8.27}$$

where the superscript "new" indicates the adjusted value, and ε_r^d is the desired relative truncation error (the default value is $1 \cdot 10^{-6}$). This algorithm performs a reasonable job of maintaining the desired truncation error. User-specified minimum- and maximum-allowed sub-time step values prevent unrealistic sub-time steps. Also, the sub-time step is not allowed to exceed the system time step.

A variable of general interest is the reactor period, which can be considered as the time constant for the reactor power. The following inverse period is calculated and provided in the reactor dynamics output:

$$\frac{1}{\tau} = \omega = \frac{\ln\left(\dfrac{n^{m+1}}{n^{m}} \right)}{\Delta t_r} \tag{8.28}$$

where τ is the reactor period based on the total neutron power, and ω is the corresponding inverse period. Also provided is

$$\frac{1}{\tau_e} = \omega_e = \frac{\ln\left(\dfrac{n_e^{m+1}}{n_e^{m}} \right)}{\Delta t_r} \tag{8.29}$$

where τ_e is the reactor period based on the effective thermal power, and ω_e is the corresponding inverse period. The reactor period is sometimes called the asymptotic period or the e-folding time.

Reactivity, as defined by Equation (8.3), is a unitless quantity. SAFSIM also provides the reactivity in units of dollars ($) and cents (¢) where $1 of reactivity is defined as a reactivity equal to the effective total delayed neutron fraction. A reactivity equal to half of this is then referred to as 50¢ of reactivity. Expressed in equation form:

$$\rho_\$ = \frac{\rho}{\beta} \tag{8.30}$$

where $\rho_\$$ is the reactivity in dollars and cents.

The reactor dynamics model in SAFSIM is general in that the analyst has considerable control over the problem specifications. In addition to allowing selection of the number of delayed neutron and decay heat precursor groups, the number of reactors can be selected. Because any number of feedback reactivities can be arbitrarily specified, multiple reactors can be coupled by providing appropriate coupling feedback terms for the different reactors. Such an approach is consistent with the engineering philosophy adopted for all of SAFSIM's models.

9.0 Function-Controlled Variables and Functions

Function-controlled variables are a special feature provided in SAFSIM thet allow the analyst to specify many of the input variables as any function of many of the output variables. Though there is really no theory associated with function-controlled variables, this section is included to provide a few qualitative examples of the versatility they allow in modeling complex systems. Reference 24 contains quantitative examples of functions and function-controlled variables along with additional details on their use and implementation.

Function-controlled variables and functions are implemented in SAFSIM in such a fashion that if they are not used, they impose no computational overhead. The basic forms for a function-controlled variable are

$$y = x \tag{9.1}$$

and

$$y = f(x) \tag{9.2}$$

where y is the function-controlled variable, x is a signal-variable function, and a mathematical function is indicated by f( ). Functions can be nested such that they are functions of other functions. A signal variable function is a basic output variable from SAFSIM, such as temperature or pressure. A mathematical function is any function such as cosine (and other trigonometric functions), table lookup, conditional statements, max and min functions, polynomials, averaging, and radiative heat fluxes. The equations for a few mathematical functions are provided in Appendix D. A list of variables that can be function controlled along with a library of available functions is provided in Appendices A and B of Reference 24. Analysts are encouraged to review these lists to become familiar with what is available.

Functions and function-controlled variables are updated either at the beginning or end of the system time step. Placement of the function and function-controlled variable input data allows selection of when they are updated: after solution of the fluid mechanics, structure heat transfer, or reactor dynamics physics modules. A few examples of the use of function-controlled variables are offered here to provide a spring board for their development for specific problems. Only simple qualitative examples are offered because the details of how a particular function-controlled variable (and associated functions) is implemented depends on the specific application.

Example 1: The action of a valve can be simulated by specifying the K-factor for a fluid mechanics finite element to be a function of some signal. A check valve can be simulated by specifying the signal as the valve pressure drop. Also, a motor operated valve can be simulated by specifying the signal as time. Inertial effects of the valve can be included in the function if desired.

Example 2: The effect on friction factor due to thermal expansion and contraction of the walls of an annulus can be simulated by specifying the equivalent diameter of the appropriate fluid mechanics finite elements as a function of the temperature of the coupled structure heat transfer finite elements. The mathematical function can be a simple thermal expansion model or a more complex stress-strain-displacement model.

Example 3: A temperature or heat flux boundary condition for a structure can be specified as a cosine function of time to simulate an oscillatory boundary. Conditional statements can be included to switch the function to a linear ramp based on the occurrence of some other event, such as the attainment of a given mass flow rate.

Example 4: Very complex flow systems can be separated into several individual networks to increase program execution speed (by decreasing the semi-bandwidth of the associated matrix equations). The connecting nodes of the networks can be linked via function-controlled variable boundary conditions.

Example 5: Reactivity feedback terms such as Doppler feedback can be specified as a function of an average reactor fuel temperature. All feedback terms must be specified using function-controlled variables.

Example 6: Exchange surfaces of multiple heat transfer structures can be radiatively coupled, including view factors. Likewise the surfaces can be conductively coupled, including contact resistances. Pseudo multidimensional conduction models also can be built in this way.

Example 7: Time step control for the system can be specified as a function of the behavior of its components. Likewise, convergence criteria can be controlled by factors such as the attainment of maximum or minimum limits. Also, the choice of quasi-static or partially dynamic fluid mechanics solution procedures can be specified as a function of the time rate of change of certain variables.

Example 8: The Reynolds number for transition between laminar and turbulent flow heat transfer can be used as an on-off flag to simulate dryout (transition from liquid to vapor) of a structure surface.

Example 9: Pump speed and direct fluid heating can be controlled to simulate predicted pump cavitation.

Example 10: Flow areas and diameters of fluid mechanics finite elements can be specified as function of wall temperature and fluid velocity to account for ablation of the surface.

As is evident, the possibilities are essentially endless. And for very complex functions, the analyst can make use of the user-supplied function interface. This allows analysts to incorporate their own functions into SAFSIM if desired. The analyst must be cautious not to implement a feedback loop that is incompatible with the numerical solutioı strategies used in SAFSIM; it is possible to put the program into an infinite loop.

It should be pointed out that function-controlled variables that are specified as functions of time are modeled in a quasi-static mode. Thus the time-derivative of the variable is neglected in the governing equations. For example, the fluid mechanics governing equations are formulated using a fixed control volume. Therefore, the geometry time-derivative terms are neglected. As long as theses terms are small compared to other terms in the equations, neglecting them is appropriate. Analysts must be aware that this may not always be the case and so using function-controlled variables to simulate time-varying geometry would then be inappropriate.

Another point that must be made is that if, for example, volume is specified as a function-controlled variable, related quantities such as flow area and length are not automatically modified. This is because there is more than one way to make the modification (such as maintaining a constant area and changing the length or vise versa). Therefore, if changes in related quantities significantly affect the solution, the analyst must also specify the related quantities as function-controlled variables using appropriate functional relationships.

10.0 System Solution Procedures

In the previous sections, the governing equations and the associated solution procedures for the three physics modules are described. This section contains a description of the steady-state and transient solution procedures used for the system. A system is any combination of fluid mechanics flow networks, heat transfer structures, nuclear reactors, and associated function-controlled variables and functions. The system solution procedures couple the three physics modules and the function-controlled variables to allow simulation of the synergistic effects between components.

The coupling of the physics modules in SAFSIM is explicit. This means that the latest output values from one physics module are used as input for the next physics module. There is no iteration between the modules to advance all physics module solutions to a consistent end of time step solution. The disadvantage of this approach is that for very tightly coupled systems, smaller system time steps are required compared to an implicitly coupled procedure. Thus multiple system level iterations are avoided in an explicit procedure at the expense of smaller system time steps. In general, the uncertainty associated with explicit coupling is small compared to other uncertainties in a system model, such as those associated with material properties, geometry, heat transfer coefficients, and friction factors.

There are advantages, however, to an explicitly coupled procedure. In addition to being easy to implement, such a procedure allows the specification of different sub-time steps for each fluid mechanics network, heat transfer structure, and nuclear reactor in the system, all of which may have vastly different characteristic time constants. Explicit coupling thereby improves computational efficiency because a component with a relatively slow time response is not forced to execute at the smaller time step associated with a fast-response component. The analyst can always reduce the system time step until the uncertainties associated with explicit coupling are sufficiently small for the problem of interest.

Figure 12 provides a schematic of the system solution procedure for both steady-state and transient simulations. The fluid mechanics module is updated first for all flow networks. Next, the structure heat transfer module is updated for all structures. Finally, the reactor dynamics module is updated for all nuclear reactors. The figure also shows the order that the function-controlled variables and functions are updated for the system and for the fluid mechanics, structure heat transfer, and reactor dynamics physics modules. It is important to understand that any function-controlled variable can be placed in any of the function-controlled variable blocks.✦ The same is true for functions. Thus placing a function in the block for fluid mechanics specifies that the function is evaluated immediately after the fluid mechanics modules are updated. Also note that functions are evaluated before function-controlled variables. Generally, functions and function-controlled variables that are dependent on time should be placed in the system block because they are updated at the beginning of each new system time step, before any of the physics modules are updated.

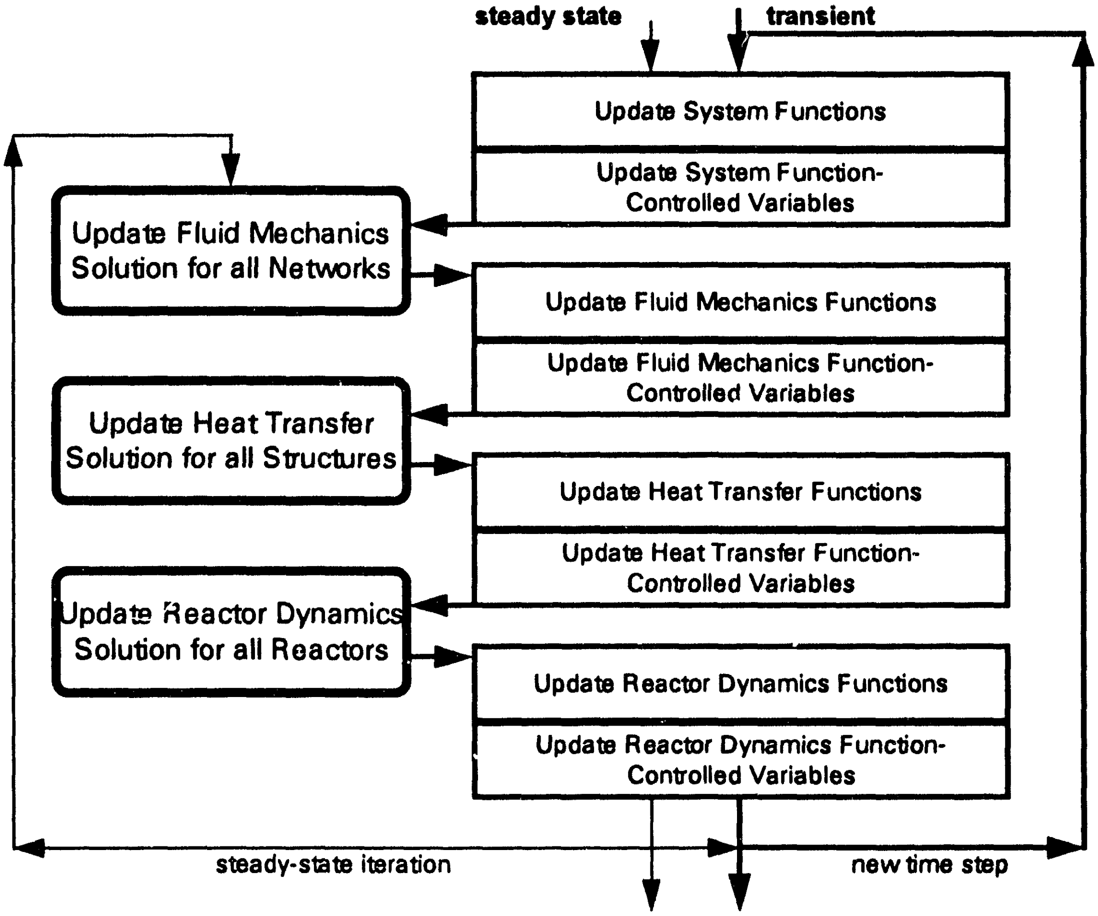

steady state
transient
Update System Functions
Update System Function-Controlled Variables
Update Fluid Mechanics Solution for all Networks
Update Fluid Mechanics Functions
Update Fluid Mechanics Function-Controlled Variables
Update Heat Transfer Solution for all Structures
Update Heat Transfer Functions
Update Heat Transfer Function-Controlled Variables
Update Reactor Dynamics Solution for all Reactors
Update Reactor Dynamics Functions
Update Reactor Dynamics Function-Controlled Variables
steady-state iteration
new time step

Figure 12. Schematic of the System Solution Procedure

Two paths are shown in Figure 12, one path for steady-state simulations and one path for transient simulations. Note that for steady-state simulations, functions and function controlled variables in the system block are updated only once at the beginning of execution based on the user-specified initial conditions. For example, if a mass flow rate boundary condition is specified as a function of time in the system block, it is evaluated at the problem start time (*variable*

✦ SAFSIM input is divided into eighteen data blocks. Eight of these blocks are for function-controlled variables and functions. Refer to the input manual, Reference 24, for details.

TSTART) specified by the analyst. It is not updated again during the steady-state simulation. If a transient simulation is then initiated subsequent to the steady-state simulation, the mass flow rate boundary condition is updated at the beginning of every new time step. Modeling versatility is enhanced by allowing the analyst to select the placement for functions and function-controlled variables; analysts should evaluate the optimum placements based on the specifics of each problem.

As mentioned earlier, the physics modules, function-controlled variables and functions are explicitly coupled for transient simulations. Actually, the coupling is not purely explicit. For example, the temperature solution for a heat transfer structure that includes convection to a fluid network uses the updated fluid temperatures (end of current time step values). This is because the fluid mechanics modules are updated before the structure heat transfer modules. On the other hand, the temperature solution for the coupled fluid network uses the structure temperatures from the end of the previous time step because those values are the most current available. Such coupling is sometimes referred to as semiexplicit, which may connote that there are limited iterations between modules. The coupling is referred to as explicit in this manual to emphasize that there are no iterations between modules.

Two methods are available in SAFSIM to perform steady-state simulations. The first method involves executing a transient simulation until all time-derivative terms are sufficiently small. Function-controlled variables can be configured to monitor such convergence if desired. The second method is, in general, faster but may not be as robust for very complicated systems. In the second method, all time-derivative terms are set to zero. Iterations are then performed between physics modules, function-controlled variables and functions until the wall temperatures for all convective exchange surfaces converge. A relaxation parameter can be specified according to the following equation:

$$T_{w,R}^{i} = \lambda_w T_w^i + (1-\lambda_w) T_w^{i-1} \tag{10.1}$$

where $T_{w,R}^{i}$ is the relaxed wall temperature for the current iteration i, λ_w is the wall temperature relaxation parameter, T_w^i is the wall temperature for the current iteration before relaxation, and T_w^{i-1} is the wall temperature for the previous iteration i-1. Generally, overrelaxation ($\lambda_w \approx$ 1.6) greatly accelerates convergence to steady state.

Convergence is monitored based on both absolute and relative wall temperature errors. The absolute error is

$$\varepsilon_w^A = \left| T_w^i - T_w^{i-1} \right| \tag{10.2}$$

and the relative error is

$$\varepsilon_w = \frac{\left| T_w^i - T_w^{i-1} \right|}{T_w^i} = \frac{\varepsilon_w^A}{T_w^i} \qquad (10.3)$$

The relative error convergence is checked only if the absolute error is not satisfied. This facilitates convergence for networks containing both high- and low-temperature structures. For systems with no convective exchange surfaces, three coupling iterations are taken. This is usually sufficient to achieve consistency between physics modules and function-controlled variables. Following steady-state convergence, a transient simulation is automatically initiated if the problem end time (*variable* TEND) is greater than the start time. Thus the two steady-state simulation methods can be combined if desired or if necessary for complex systems.

As discussed in Section 6, quasi-static, partially dynamic, and fully dynamic solution modes can be implemented for transient fluid mechanics simulations. Also, quasi-static and dynamic solution modes can be implemented for transient structure heat transfer simulations. The desired mode for each flow network and each structure is specified in the input and can be function controlled. Thus the analyst can change modes during the course of a transient to enhance execution speed. Whether the mechanical energy equation is solved can also be function controlled. Turning the mechanical energy equation off may be desirable if the flow field is not changing significantly. Choosing the solution mode and mechanical energy equation option also allows the analyst to perform parametric "what if" analyses to improve understanding of a system's response.

Throughout this manual, several user-supplied subroutines are discussed that allow the analyst to interface with the physics modules and functions. A system level subroutine interface is also available to further enhance the ability to interact with SAFSIM. This subroutine is called after all physics modules, function-controlled variables, and functions are updated for every steady-state iteration and for every transient time step.

11.0 Summary and Comments

This section provides a brief summary of SAFSIM. In addition, several comments are offered regarding its development and use. These comments provide additional guidance on the application of SAFSIM to system modeling.

SAFSIM was developed to provide a general-purpose engineering tool for simulating the integrated performance of complex flow systems. Extensive use of engineering factors and approximate methods are made to simplify complex systems to a level appropriate for fast execution on personal computers and workstations. This approach is common in the engineering sciences when the component-level details are not as important as the overall system performance and when approximate engineering solutions are needed with minimal expenditure of resources.

Fluid mechanics is modeled using a 1-D network finite element method. Structure heat transfer also is modeled using a 1-D finite element method with multiple convection heat transfer exchange surface capability to enable coupling to the fluid mechanics solution. A point reactor dynamics model is included to allow simulation of nuclear reactors, including feedback coupling to the fluid mechanics and structure heat transfer solutions.

Although the fluid mechanics model is based on 1-D geometry, networks of finite elements connected in series and parallel can be constructed. The structure heat transfer model also is 1-D, but flow networks and structures can be coupled via multiple exchange surfaces. Together, the two models provide a pseudo-multidimensional capability. Such extended 1-D models are often sufficient to obtain a first-look understanding of a system's performance and provide guidance as to where to spend additional computational and experimental analysis resources.

The reactor dynamics model is included in SAFSIM to allow simulation of nuclear reactors for power and propulsion applications. In fact, SAFSIM was developed specifically to address the system response of several conceptual advanced space nuclear reactor systems and related experiments. However, SAFSIM's versatility enables its application to a broad range of engineering disciplines outside the nuclear field. Also, this versatility makes SAFSIM particularly useful for performing parametric analyses.

Developing input models can be very time consuming because all geometric, property, and operational information must be provided by the analyst. In fact, for complex systems, the model development time may exceed the simulation and analysis time. Although SAFSIM is based primarily on 1-D physics models, creating input models can be quite challenging. In many instances, creating a meaningful 1-D input model is more difficult than creating a multidimensional model. This is because a significant amount of engineering judgment and knowledge is required to condense the actual multidimensional system into an equivalent 1-D representation. Because there is more than one way to perform this condensation, more than one model may need to be constructed to determine which is most appropriate for the problem under consideration.

There are many graphics-based computer programs available for generating computational meshes for multidimensional geometries. Unfortunately, there are no graphical computer programs available to create 1-D input models from detailed geometry information. Such a program, complete with an engineering knowledge data base, is a long-term development goal for SAFSIM. Until such a program exists, it is necessary for the analyst to transform the problem description into the appropriate form for SAFSIM input. For complex problems, or for problems in which geometric design sensitivities are to be studied, it may be worthwhile to create special-purpose programs to facilitate the input generation process.

Several suggestions are offered here to aid the analyst in the process of condensing complex multidimensional data to equivalent 1-D representations. Suggestions also are included to improve the computational efficiency of the simulations.

- Identical parallel flow channels can be grouped into a single equivalent channel if the details of the flow in each of the separate channels are not of interest. The effective flow area and volume for the equivalent channel is the sum of the separate flow areas and volumes. The flow length and equivalent diameter, however, are the same as that for the identical channels in order to yield the correct Reynolds and Peclet numbers, and friction factor.

- Similar pumps connected in series or in parallel can be grouped into a single equivalent pump. For multiple pumps connected in series, a combined characteristic curve is formed by adding the pump heads for each volumetric flow rate value. For multiple pumps connected in parallel, a combined characteristic curve is formed by adding the volumetric flow rates for each pump head value.

- Very complex flow networks can be separated into smaller, simpler networks to reduce the semibandwidth of the global pressure matrices and thereby reduce potential roundoff error. The simpler networks are coupled via function-controlled pressure and mass flow rate boundary conditions. Thus the outlet pressure boundary condition of one of the simpler networks is specified as the calculated pressure for the inlet of a connecting simpler network. The inlet mass flow rate boundary condition of the connecting network is then specified as the calculated outlet mass flow rate of the other network. Because function-controlled variables are evaluated only after a system time step, the system time step might have to be reduced, compared to the single complex network, to provide adequate coupling of the simpler networks.

- Transients in fluid flow networks involving gases can often be adequately simulated in quasi-static mode to increase computational efficiency. This mode is an option in SAFSIM whereby the time-derivative terms are omitted from the fluid mechanics governing equations.

- Obstructions in a flow stream can be simulated, with respect to pressure drop, by including appropriate loss coefficients. Likewise, appropriately reduced wall exchange areas can be used to account for the reduction in convection heat transfer due to the obstruction. Detailed component-level simulations may be required to determine the appropriate adjustments.

- Structures with irregular shapes can be approximated with equivalent cylindrical, spherical, or rectangular structures. Volume (heat capacity) of the irregular and equivalent structures should be conserved to provide the correct total internal heat generation (if any), and to ensure reasonable dynamic response predictions. Such models provide reasonable predictions of the average temperature. Supplemental structures can be included, representing segments of the irregular structure with adjusted conduction lengths and areas, to provide estimates of peak temperatures. Detailed component-level simulations may be required to determine the appropriate adjustments for special shapes.

- Two-dimensional conduction can be approximated using multiple structures coupled via function-controlled heat flux exchange surfaces. This method of modeling conduction in a second direction should be used for the conduction direction that is not dominant.

- Correction factors can be applied using function-controlled variables to "force" a component response. The correction factor can be based on experimental data or on detailed analyses. For example, turbulent recirculation zones in a plenum can be accounted for (with respect to pressure drop) by specifying a loss coefficient as a function of flow rate. Or temperature rise in a pump can be adjusted by specifying direct fluid heating (*variable* QDIR) as a function of temperature, pressure, or mass flow rate.

- A multiple-lump, lumped-capacitance, transient conduction model can be used if temperature gradients in a certain direction are not important or not of interest. For example, a pipe wall can be constructed by choosing a rectangular geometry with 1-D conduction along the length of the pipe instead of choosing cylindrical geometry with conduction in the radial direction. Such a model provides temperature gradients along the length of the pipe but not in the radial direction, which is the direction for convection heat transfer to the fluid. Neglecting conduction in the radial direction is reasonable if the Biot number is small. (The Biot number is the ratio of the thermal resistance within the solid to the resistance from fluid convection at the surface.) This approach is computationally faster than using multiple structures with radial conduction because only a single structure is needed for the entire pipe. If the Biot number is large (greater than about 0.1), this approach can still be used if the heat transfer coefficient is modified to account for the large internal conductive resistance of the solid. A technique for determining such modified heat transfer coefficients for cylindrical fuel rods is provided in Reference 35. This technique was generalized for multilayered cylindrical structures and the resulting equations are provided in Appendix E.

- A multiple-lump, lumped-capacitance, transient conduction model can also be used for packed beds. In such a model, heat transfer within the bed is based on an effective bed conductivity that accounts for particle-to-particle thermal radiation and conduction at the contact points. A modified heat transfer coefficient is then used to account for the internal conductive resistance of the particles comprising the packed bed. A technique for determining such modified heat transfer coefficients for spherical fuel particles is provided in Reference 36. This technique was generalized for multilayered spherical structures and the resulting equations are provided in Appendix E.

- Identical heat transfer structures can be grouped by simply modeling one structure and specifying the appropriate number of copies (*variable* COPIES) to provide the correct total power convected to the fluid.

Because of the complexities involved in flow system modeling, the analyst should check the computed results for reasonableness. Sometimes it might be desirable to isolate a part of the system to verify that its performance is as expected. As with any numerical model, mesh size parametric studies should be conducted to verify that the results are not too strongly dependent on the chosen mesh. Similarly, time step parametric studies should be conducted for transient simulations, especially for the system time step. Whenever possible, the predicted results should be checked with experimental results, other computer programs, or even with simple analytic equations. An example of such a check is to perform a simple steady-state energy balance on the fluid over all or selected regions of the system. It should be noted that because SAFSIM provides the solution to a total energy equation, kinetic and potential energy must be included in such a balance. Thus

$$q_t = \dot{m}\left[\overbrace{\underbrace{(h_2 - h_1)}_{\substack{\text{internal and}\\\text{pressure energy}}} + \underbrace{\tfrac{1}{2}\left(v_2^2 - v_1^2\right)}_{\substack{\text{kinetic}\\\text{energy}}} + \underbrace{g_z(z_2 - z_1)}_{\substack{\text{potential}\\\text{energy}}}}^{\text{total energy}}\right] \tag{11.1}$$

where q_t is the total heating rate to the fluid (power), $\dot{m}$ is the mass flow rate, h is the fluid enthalpy, v is the fluid velocity, g_z is the acceleration due to gravity in the vertical direction, and z is the elevation. The 1 and 2 subscripts indicate the entrance and exit of the selected region, respectively.

The current version of SAFSIM contains about 20,000 FORTRAN lines; about one-third of the lines are descriptive comment lines. SAFSIM development is by no means complete. However, the program architecture was designed with expansion and modification in mind; thus new features and capabilities can be added with relative programming ease. Also, SAFSIM was designed with many user interfaces built into the program, allowing each analyst to tailor SAFSIM for their particular application without having to fully understand the internal programming structure and solution techniques.

A sample application is provided in Appendix F to demonstrate the type of problems that can be addressed with SAFSIM. The problem addresses the laminar flow instability of an internally heated packed bed cooled with a gas. Input and output files are included.

Versatility is the primary development objective of SAFSIM provided by numerous input options and modeling features. Because of the many available options, it is impossible to check all possible combinations of options that an analyst may select. Also, not every SAFSIM feature has been fully tested or implemented. Therefore, although a limited number of benchmarking and verification problems have been exercised, the analyst should conduct their own problem-specific benchmarking whenever possible. As is the case with any general-purpose computer program under development, "bugs" should be expected.

SAFSIM is a functional computer program that runs on a personal computer or workstation and provides the analyst with a tool for obtaining engineering solutions to complex system analysis problems. The general-purpose nature and versatility of SAFSIM make it a candidate analysis tool for a broad range of engineering disciplines. Additional enhancements and features are envisioned to make SAFSIM even more versatile, expanding the class of problems for which it is applicable.

Seiver's Law of Simulation:

> *Every Simulation is Accompanied by an Equal Amount of Dissimulation*

An Addendum to Seiver's Law:

> *Distinguishing the Dissimulation From the Simulation is the Engineer's Goal*

12.0 References

1. Ransom, V. H., R. J. Wagner, et al., RELAP5/MOD1 Code Manual Volume 1: System Models and Numerical Methods, NUREG/CR-1826, 1981.

2. Fox, Robert W., and Alan T. McDonald, *Introduction to Fluid Mechanics*, John Wiley & Sons, New York, NY, Third Edition, 1985.

3. White, Frank M., *Fluid Mechanics*, McGraw-Hill Inc., New York, NY, Second Edition, 1986.

4. Panton, Ronald L., *Incompressible Flow*, John Wiley & Sons, New York, NY, 1984.

5. Bird, Byron R., Warren E. Stewart, and Edwin N. Lightfoot, *Transport Phenomena*, John Wiley & Sons, New York, NY, 1960.

6. Idelchik, I. E., *Handbook of Hydraulic Resistance*, Hemisphere Publishing Corporation, New York, NY, Second Edition, 1986.

7. Moody, L. F., "Friction Factors for Pipe Flow,"*ASME Trans.*, Vol. 66, pp. 671-684, 1944.

8. Swamee, P. K., and A. K. Jain, "Explicit Equations for Pipe-Flow Problems," *J. Hydr. Div., Proc. ASCE*, pp. 657-664, May 1976.

9. Petukhov, B. S., "Heat Transfer and Friction in Turbulent Pipe Flow with Variable Physical Properties," in *Advances in Heat Transfer*, Academic Press, New York, NY, Vol. 6, p. 504, 1970.

10. Ergun, S., "Fluid Flow Through Packed Columns," in *Chem. Eng. Progress*, Vol. 48, No. 2, pp. 89-94, 1952.

11. Macdonald, I. F., M. S. El-Sayed, K. Mow, and F. A. L. Dullien, "Flow through Porous Media – the Ergun Equation Revisited," in *Ind. Eng. Chem. Fundam.*, Vol. 18, No. 3, pp. 199-208, 1979.

12. Achenbach, E., "Heat Transfer and Pressure Drop of Pebble Beds Up to High Reynolds Number," in *Proceedings of Seventh International Heat Transfer Conference*, Munchen, Federal Republic of Germany, Vol. 1, pp. 3-8, 1982.

13. Howell, John, R., and Richard O. Buckius, *Fundamentals of Engineering Thermodynamics*, McGraw-Hill Book Co., New York, NY, 1987.

14. White, Frank M., *Viscous Fluid Flow*, McGraw Hill Book Co., New York, NY, 1974.

15. Reid, Robert C., John M. Prausnitz, and Bruce E. Poling, *The Properties of Gases and Liquids*, McGraw-Hill Book Co., New York, NY, Fourth Edition, 1987.

16. Walkao, N. and S. Kaguei, *Heat and Mass Transfer in Packed Beds*, Gordon and Breach Science Publishers, New York, NY, 1982.

17. Huebner, Kenneth H., and Earl A. Thornton, *The Finite Element Method for Engineers*, John Wiley & Sons, New York, NY, Second Edition, 1985.

18. Becker, Eric B., Graham F. Carey, and J. Tinsley Oden, *Finite Elements - An Introduction Volume 1*, Prentice-Hall, Inc., Englewood Cliffs, New Jersey, 1981.

19. Rohsenow, Warren M., James P. Hartnett, and Ejup N. Ganic, *Handbook of Heat Transfer Fundamentals*, McGraw-Hill Book Co., New York, NY, Second Edition, 1985.

20. Sheppard, Susan J. and John E. McAllister, Jr., "Pressure Drop Correlation for Blowing Conditions in a Cylindrical Geometry," in *Proceedings of Nuclear Technologies for Space Exploration Conference*, NTSE92, Vol. 2, pp. 549-558, August 1992.

21. Dobranich, Dean, "Some Parametric Fluid Flow Analyses of a Particle Bed Element Using SAFSIM," AIAA/SAE/ASME/ASEE 29th Joint Propulsion Conference, paper # AIAA 93-1756, June 1993.

22. Patankar, S. V., *Numerical Heat Transfer and Fluid FLow*, Hemisphere/McGraw-Hill, New York, NY, 1980.

23. Chapra, Steven C., and Raymond P. Canale, *Numerical Methods for Engineers*, McGraw-Hill Book Co., New York, NY, Second Edition, 1988.

24. Dobranich, Dean, SAFSIM Input Manual -- A Computer Program for the Engineering Simulation of Flow Systems, SAND92-0694, Sandia National Laboratories, Albuquerque, NM, September 1992.

25. Koh, J. C., J. L. Dutton, B. A. Benson, and A. Fortini, "Friction Factor for Isothermal and Nonisothermal Flow Through Porous Media," in *Journal of Heat Transfer*, Vol. 99, August 1977.

26. Chapman, Alan J., *Fundamentals of Heat Transfer*, Macmillan Publishing Company, New York, NY, 1987.

27. Holman, J. P., *Heat Transfer*, McGraw-Hill Book Co., New York, NY, Sixth Edition, 1986.

28. Patankar, S. V., and B. R. Baliga, "A New Finite-Difference Scheme for Parabolic Differential Equations," in *Numerical Heat Transfer*, Vol. 1, pp. 27-37, 1978.

29. Schotte, William, "Thermal Conductivity of Packed Beds," in *A.I.Ch.E. Journal*, Vol. 6, No. 1, pp.63-67, March 1960.

30. Ott, Karl O., and Robert J. Neuhold, *Nuclear Reactor Dynamics*, American Nuclear Society, Chicago, IL, 1985.

31. Hetrick, David L., *Dynamics of Nuclear Reactors*, University of Chicago Press, Chicago, IL, 1971.

32. Keepin, G. R., *The Physics of Nuclear Kinetics*, Addison Wesley Publishers, Reading, MA, 1965.

33. *Proposed ANS Standard Decay Energy Release Rates Following Shutdown of Uranium-Fueled Thermal Reactors*, ANS-5.1, American Nuclear Society, Chicago, IL, 1971.

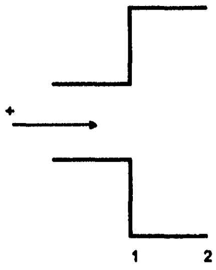
34. Kagonove, J. J., Numerical Solution of the One-Group Space-Independent Reactor Kinetics Equations for Neutron Density Given the Excess Reactivity, ANL-6132, Argonne National Laboratory, Argonne, IL, 1960.

35. Meyer, John E., Some Physical and Numerical Considerations for the SSC-S Code, NUREG/CR-0451, BNL-NUREG-50913, Brookhaven National Laboratory, NY, September 1978.

36. Tuddenham, Read Stapley, "Thermal Hydraulic Analysis of a Packed Bed Reactor Fuel Element," Masters Thesis, Massachusetts Institute of Technology, Boston, MA, May 1989.

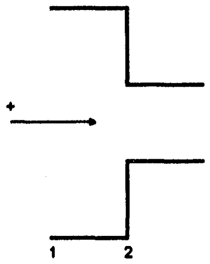
Appendix A: Area Change Loss Coefficients

The following equations[A1] are used in SAFSIM, if desired, to automatically determine appropriate loss coefficients (*K*-factors) for abrupt and gradual expansions and contractions. The positive flow direction is from position 1 to position 2 as indicated by the arrows. Angles are in degrees.



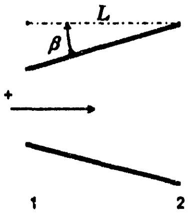
Abrupt Expansion (based on velocity at position 1)

$$K_{1-2} = 1.05\left(1 - \frac{2A_1}{3A_2}\right) + \left(\frac{A_1}{A_2}\right)^2 - \frac{4A_1}{3A_2} \qquad \text{(A.1)}$$



Abrupt Contraction (based on velocity at position 2)

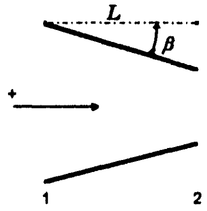
$$K_{1-2} = \frac{1}{2}\left(1 - \frac{A_2}{A_1}\right)^{\frac{3}{4}} \qquad \text{(A.2)}$$



Gradual Expansion (based on velocity at position 1)

$$K_{1-2} = 3.2[\tan\beta]^{1.25}\left(1 - \frac{A_1}{A_2}\right)^2 \text{ where } [\tan\beta] = \frac{\sqrt{A_2/A_1} - 1}{L}\sqrt{\frac{A_1}{\pi}}$$

(A.3)



Gradual Contraction (based on velocity at position 2)

$$K_{1-2} = \left[-0.0125\left(\frac{A_2}{A_1}\right)^4 + 0.0224\left(\frac{A_2}{A_1}\right)^3 - 0.00723\left(\frac{A_2}{A_1}\right)^2 + 0.00444\left(\frac{A_2}{A_1}\right) - 0.00745\right]\left[\varphi^3 - 2\pi\varphi^2 - 10\varphi\right] \text{ where } \varphi = 0.0349\beta$$

(A.4)

Figure A1. Area Change Loss Coefficients

The following criteria are used to distinguish between abrupt and gradual expansions and contractions:

$$\text{if } \left|\frac{A_1 - A}{A}\right| > 2.5\left|\frac{A_2 - A}{A}\right| \text{ then the expansion is abrupt}$$

$$\text{if } \left|\frac{A_2 - A}{A}\right| > 2.5\left|\frac{A_1 - A}{A}\right| \text{ then the contraction is abrupt} \tag{A.5}$$

where A is the finite element area, and A_1 and A_2 are the areas at the nodes. These criteria were arbitrarily chosen for SAFSIM implementation and may not be appropriate for all situations. Equations (A.1) through (A.4) are applicable to reverse flow simply by reversing the 1 and 2 positions (i.e., A_1 becomes A_2 and A_2 becomes A_1).

Appendix A References

A1. Idelchik, I. E., *Handbook of Hydraulic Resistance*, Hemisphere Publishing Corporation, New York, NY, Second Edition, pp. 145-215, 1986.

Appendix B: Finite Element Assembly Procedure

The finite element assembly procedure provides the recipe by which the individual entries of the finite element local matrices are combined to form a single global matrix. Also, the load vectors of the local finite elements are combined to form a single global load vector. If [C] is the global coefficient matrix and {b} is the global load vector, then solution of the following matrix equation provides the desired dependent variable:

$$[C]\{x\} = \{b\} \tag{B.1}$$

where {x} is the solution vector.

The assembly procedure is probably best explained with an example based on simple two-noded 1-D finite elements. An example four-element flow network is shown in Figure B1. There are five global nodes numbered 1 through 5 and the element numbers are enclosed in parentheses. The local node numbers are not included in Figure B1 to reduce clutter. However, each element has two local nodes numbered 1 and 2. The correspondence between local and global node numbers is provided in Table B1.

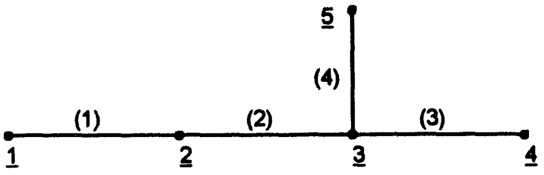


Figure B1. Example Flow Network

Table B1. Local-Global Node Number Correspondence

Element #	Local Node #	Global Node #	Local Node #	Global Node #
(1)	1	1	2	2
(2)	1	2	2	3
(3)	1	3	2	4
(4)	1	3	2	5

The individual entries in a local element matrix are indicated as $c_{ik}^{(j)}$. The superscript designates the associated element number and the *ik* subscript designates the entry location in the local matrix where *i* is the row number and *k* is the column number. Thus the entries for the element (1) local matrix are

$$[c]^{(1)} = \begin{bmatrix} c_{11}^{(1)} & c_{12}^{(1)} \\ c_{21}^{(1)} & c_{22}^{(1)} \end{bmatrix} \tag{B.2}$$

Similarly, the local matrix for element (2) is

$$[c]^{(2)} = \begin{bmatrix} c_{11}^{(2)} & c_{12}^{(2)} \\ c_{21}^{(2)} & c_{22}^{(2)} \end{bmatrix} \tag{B.3}$$

Because all elements have two local nodes, the subscripts i and k are always 1 or 2. Thus global node $\underline{2}$ corresponds to local node 2 of element (1) and to local node 1 of element (2).

The individual entries for an element load vector are indicated as $b_i^{(j)}$. Again, the superscript designates the associated element number and the i subscript designates the entry location in the local load vector where i is the row number. The local load vectors for elements (1) and (2) are

$$\{b\}^{(1)} = \begin{Bmatrix} b_1^{(1)} \\ b_2^{(1)} \end{Bmatrix} \tag{B.4}$$

$$\{b\}^{(2)} = \begin{Bmatrix} b_1^{(2)} \\ b_2^{(2)} \end{Bmatrix} \tag{B.5}$$

The local element matrices and load vectors for the other three elements are similar. Following the same convention for subscripts and superscripts, the solution vector is written as

$$\{x\} = \left\{ \begin{array}{r} x_1^{(1)} \\ x_2^{(1)} = x_1^{(2)} \\ x_2^{(2)} = x_1^{(3)} = x_1^{(4)} \\ x_2^{(3)} \\ x_2^{(4)} \end{array} \right\} = \begin{Bmatrix} x_{\underline{1}} \\ x_{\underline{2}} \\ x_{\underline{3}} \\ x_{\underline{4}} \\ x_{\underline{5}} \end{Bmatrix} \tag{B.6}$$

The vector to the far right is written in terms of the global node numbers.

The first step in the assembly procedure is to construct an n by n matrix, where n is the number of global nodes (5 in this example). Next, the local element entries are inserted into the appropriate locations of the global matrix. The appropriate locations are determined by the two global node numbers for each element and the selected ordering of the solution vector entries. For element (1), the global node numbers are $\underline{1}$ and $\underline{2}$, which correspond to rows 1 and 2 of the solution vector. Thus the intersections of rows 1 and 2 with columns 1 and 2 provide the appropriate locations as indicated graphically in the next equation.

$$[C]' = \begin{array}{ccccc|l} \multicolumn{5}{c}{\text{column numbers}} & \\ 1 & 2 & 3 & 4 & 5 & \\ \hline X & X & & & & 1 \\ X & X & & & & 2 \\ & & & & & 3 \ \text{row numbers} \\ & & & & & 4 \\ & & & & & 5 \end{array} \qquad \text{(B.7)}$$

where X marks the intersection locations and the prime mark on $[C]$ indicates that this is just part of the total coefficient matrix. Following the same procedure for element (4) gives

$$[C]' = \begin{array}{ccccc|l} \multicolumn{5}{c}{\text{column numbers}} & \\ 1 & 2 & 3 & 4 & 5 & \\ \hline & & & & & 1 \\ & & & & & 2 \\ & & X & & X & 3 \ \text{row numbers} \\ & & & & & 4 \\ & & X & & X & 5 \end{array} \qquad \text{(B.8)}$$

If the entries in the solution vector are ordered differently, different intersection locations occur but the procedure is the same. The diagonal entries of each of the local element matrices are then inserted into the diagonal intersection locations of the global matrix such that each entry multiplies the appropriate x entry. Likewise, the lower and upper offdiagonal entries of the local element matrices are inserted into the lower and upper offdiagonal entries of the global matrix, respectively. (Upper offdiagonal entries are those above the diagonal entries.)

Assembling the local matrices for the five finite elements produces the following global matrix:

$$[C] = \begin{bmatrix} c_{11}^{(1)} & c_{12}^{(1)} & 0 & 0 & 0 \\ c_{21}^{(1)} & \left(c_{22}^{(1)} + c_{11}^{(2)}\right) & c_{12}^{(2)} & 0 & 0 \\ 0 & c_{21}^{(2)} & \left(c_{22}^{(2)} + c_{11}^{(3)} + c_{11}^{(4)}\right) & c_{12}^{(3)} & c_{12}^{(4)} \\ 0 & 0 & c_{21}^{(3)} & c_{22}^{(3)} & 0 \\ 0 & 0 & c_{21}^{(4)} & 0 & c_{22}^{(4)} \end{bmatrix} \qquad \text{(B.9)}$$

and the associated global load vector is

$$\{b\} = \begin{Bmatrix} b_1^{(1)} \\ b_2^{(1)} + b_1^{(2)} \\ b_2^{(2)} + b_1^{(3)} + b_1^{(4)} \\ b_2^{(3)} \\ b_2^{(4)} \end{Bmatrix} \tag{B.10}$$

If x is specified as a boundary condition for a global node, then the corresponding column from the coefficient matrix is multiplied by the corresponding row of the solution vector and moved into the load vector; the corresponding row is then eliminated, thereby reducing the order of the matrix by 1. In the example, if x_1 is specified, the load vector becomes (before elimination of the first row)

$$\{b\} = \begin{Bmatrix} b_1^{(1)} - x_1 c_{11}^{(1)} \\ b_2^{(1)} + b_1^{(2)} - x_1 c_{21}^{(1)} \\ b_2^{(2)} + b_1^{(3)} + b_1^{(4)} \\ b_2^{(3)} \\ b_2^{(4)} \end{Bmatrix} \tag{B.11}$$

Now the first row is omitted from all global vectors and from the coefficient matrix to give the following matrix equation:

$$\begin{bmatrix} \left(c_{22}^{(1)} + c_{11}^{(2)}\right) & c_{12}^{(2)} & 0 & 0 \\ c_{21}^{(2)} & \left(c_{22}^{(2)} + c_{11}^{(3)} + c_{11}^{(4)}\right) & c_{12}^{(3)} & c_{12}^{(4)} \\ 0 & c_{21}^{(3)} & c_{22}^{(3)} & 0 \\ 0 & c_{21}^{(4)} & 0 & c_{22}^{(4)} \end{bmatrix} \begin{Bmatrix} x_2 \\ x_3 \\ x_4 \\ x_5 \end{Bmatrix} = \begin{Bmatrix} b_2^{(1)} + b_1^{(2)} - x_1 c_{21}^{(1)} \\ b_2^{(2)} + b_1^{(3)} + b_1^{(4)} \\ b_2^{(3)} \\ b_2^{(4)} \end{Bmatrix} \tag{B.12}$$

In the preceeding procedure, the global matrix was assembled element by element. It is also possible to form the global matrix one row at a time. For a given row i, the global diagonal entry is just the sum of the local diagonal entries for all elements connected to node i. If the element is connected at local node 1 then the c_{11} entry is used and similarly, if the element is connected at local node 2 then the c_{22} entry is used. If global node i is a boundary node, the appropriate c entry times x_i is subtracted from the load vector entry for that row. The lower global offdiagonal entries are simply equal to the lower local off-diagonal entries of all connecting elements and the upper global offdiagonal entries are equal to the upper local offdiagonal entries of all connecting elements.

Such a row-by-row procedure requires information about the number of elements connected to each global node, whether the connection is at local node 1 or 2, and whether a boundary node is involved. Also, flow direction must be accounted for if upwind factors are involved. Though this procedure is difficult to describe succinctly in words, it is "cookbook" and therefore relatively straightforward to program. The row-by-row procedure is implemented in SAFSIM.

A very nice feature of the two assembly procedures is that they are applicable to all types of elements, including higher-order elements and multidimensional elements. Thus the assembly procedure remains unchanged if the types of elements are changed or if different elements are added. Although a significant amount of bookkeeping must be performed, the procedures offer the advantage of versatility.

A few words about global node numbering and bandwidth minimization are appropriate at this time. An important feature of a matrix is its bandedness, which is the degree to which the nonzero entries of the matrix are banded about the diagonal. Keeping the matrix entries close to the diagonal is beneficial with respect to computational efficiency and storage requirements. Smaller semibandwidth matrices require less computations and produce less roundoff error. A useful measure of bandedness is the semibandwidth, which is equal to the maximum number of upper offdiagonal, nonzero entries of all rows plus 1. Referring to Equation (B.9) of the example, the semibandwidth is determined by row three and is equal to 3. Row three has 2 upper offdiagonal entries. (All other rows have but 1 upper offdiagonal entry.) Therefore, 2 + 1 = 3 is the semibandwidth.

A simple way to determine the semibandwidth of a matrix is to take the difference between the two global node numbers of each finite element. The maximum difference plus 1 is then the semibandwidth. Again, referring to the example shown in Figure B1, Element (4) has a difference of $\underline{5} - \underline{3} = 2$. Adding 1 to this maximum difference produces 3. Thus the node numbers should be assigned such that the maximum difference for any two adjacent nodes is minimized. In the example, if node numbers $\underline{1}$ and $\underline{3}$ are interchanged, the maximum difference is $\underline{5} - \underline{1} = 4$; this results in the more unfavorable semibandwidth of 5. In general, the global node numbers for solution of the mechanical energy equation should be chosen to minimize the semibandwidth.

Intentionally Blank

Appendix C: Selected Heat Transfer Coefficients

In SAFSIM, the analyst is provided with a library of 85 heat transfer coefficient correlations from which to choose. Two correlations must be specified for each convection heat transfer exchange surface; one for laminar flow and one for turbulent flow. The entire list of available correlations is provided in Appendix C of the SAFSIM input manual,[C1] along with the associated references and conditions for use. Analysts should consult the appropriate references to familiarize themselves with a correlation and its degree of accuracy in order to assure its proper use. Several of the more commonly used correlations are provided in this appendix as a convenience.

Equation (C.1) (correlation ID #2) provides the heat transfer coefficient[C2] for fully developed laminar flow over a constant temperature surface. It is applicable for gases or liquids with Prandtl numbers greater than 0.6. This correlation is the default in SAFSIM input for laminar flow.

$$h_c = 3.66\frac{k_t}{D} \tag{C.1}$$

where h_c is the heat transfer coefficient, k_t is the fluid thermal conductivity, and D is the equivalent diameter for heat transfer.

The correlation of Petukhov,[C3] for internal turbulent flow of a gas, is provided in Equation (C.2) (correlation ID #6). This correlation is consistent with the friction factor correlation used for smooth surfaces in the fluid mechanics solution. For rough pipes, the Moody friction factor can be substituted. The correlation is based on the Reynolds analogy in which the heat transfer coefficient is assumed proportional to the friction factor. The correlation is valid for $1{\cdot}10^4 < Re < 5{\cdot}10^6$ and $0.5 < Pr < 2000$, where Re is the Reynolds number based on the equivalent diameter for heat transfer ($\dot{m}D / A\mu$) and Pr is the Prandtl number. This correlation is the default in SAFSIM input for turbulent flow.

$$h_c = \left(\frac{k_t}{D}\right)\frac{\frac{1}{8}f\,RePr}{1.07 + 12.7\left(Pr^{\frac{2}{3}} - 1\right)\sqrt{\frac{1}{8}f}} \tag{C.2}$$

where f is the friction factor. The correlation provides 6% accuracy for Prandtl numbers between 0.5 and 200, and 10% accuracy for Prandtl numbers between 200 and 2000. Properties are evaluated at the bulk fluid temperature.

For liquids, the same correlation is used but with a correction factor applied to account for the influence of viscosity variations due to a radial temperature profile in the liquid. The liquid correlation is provided by Equation (C.3) (correlation ID #7):

$$h_c = \left(\frac{k_t}{D}\right)\frac{\frac{1}{8} f\,RePr}{1.07 + 12.7\left(Pr^{\frac{2}{3}} - 1\right)\sqrt{\frac{1}{8} f}}\left(\frac{\mu}{\mu_w}\right)^e \tag{C.3}$$

where μ_w is the viscosity evaluated at the wall temperature, and e equals 0.11 if the wall temperature is greater than the bulk liquid temperature and equals 0.25 otherwise.

One of the better-known heat transfer coefficient correlations for internal turbulent flow is the Dittus-Boelter[C3] correlation provided in Equation (C.4) (correlation ID #9). This correlation is valid for gases or liquids with $1 \cdot 10^4 < Re < 1 \cdot 10^6$ and $0.7 < Pr < 160$. It is also required that for gases, $T_w - T < 60$, and that for liquids $T_w - T < 6$, where T_w is the wall temperature.

$$h_c = \frac{k_t}{D} 0.023 Re^{0.8} Pr^e \tag{C.4}$$

where e equals 0.3 if the wall temperature is less than the bulk fluid temperature and equals 0.4 otherwise.

Equation (C.5) (correlation ID #36) provides the Achenbach[C4] correlation for flow through a packed bed, modified to account for variable porosity and Prandtl number. The correlation is valid for both laminar and turbulent flow with a Reynolds number less than $3 \cdot 10^5$. It is applicable for gases or liquids.

$$h_c = \frac{k_t}{D} 0.70731\left(\frac{1-\varepsilon}{\varepsilon}\right)\left[0.622\left(\frac{Re}{1-\varepsilon}\right)^{2.32} + 6.34 \cdot 10^{-4}\left(\frac{Re}{1-\varepsilon}\right)^3\right]^{\frac{1}{4}} Pr^{\frac{1}{3}} \tag{C.5}$$

where ε is the bed porosity, and the Reynolds number is based on the superficial flow area and an effective particle diameter.

The final correlation presented in this appendix is useful for modeling thermal radiation heat transfer from a surface. The correlation is derived by equating Newton's law of cooling to an expression for thermal radiation heat flow between two surfaces; thus

$$q_{1-2} = h_c A_1 (T_1 - T_2) = \frac{\sigma\left(T_1^4 - T_2^4\right)}{\frac{(1-e_1)}{e_1 A_1} + \frac{1}{A_1 F_{1-2}} + \frac{(1-e_2)}{e_2 A_2}} \quad \text{(C.6)}$$

where q_{1-2} is the heat flow from surface 1 to surface 2, A is the surface area, T is the surface temperature, σ is the Stefan-Boltzmann constant, e is the surface emissivity, F_{1-2} is the view factor from surface 1 to surface 2, and the 1 and 2 subscripts refer to surfaces 1 and 2, respectively. Rearrangement of this equation provides the following correlation (ID #29) for an effective radiation heat transfer coefficient:

$$h_c = \frac{\sigma}{\frac{(1-e_1)}{e_1} + \frac{1}{F_{1-2}} + \frac{(1-e_2)}{e_2}\left(\frac{A_1}{A_2}\right)}\left(T_1^2 + T_2^2\right)\left(T_1 + T_2\right) \quad \text{(C.7)}$$

Convection heat transfer exchange surfaces are coupled to the fluid of a fluid finite element. Therefore, surface 2 represents the fluid in Equation (C.7). A "dummy" fluid network can be employed with a specified exchange temperature to represent this surface. This is useful if the heat transfer from a surface to an infinite sink (such as space) is to be modeled. The dummy fluid network would then represent space with some effective temperature such as 4 K or 250 K. Note that for a convex surface enclosed by an infinite concave surface, A_1/A_2 approaches zero, F_{1-2} equals 1, and Equation (C.7) simplifies to

$$h_c = \sigma e_1\left(T_1^2 + T_2^2\right)\left(T_1 + T_2\right) \quad \text{(C.8)}$$

The surface 1 and 2 areas and emissivities, along with the view factor, are retained in Equation (C.7) for generality. Similar equations are provided as special mathematical functions to allow thermal radiation heat transfer modeling between two structure surfaces via function-controlled variables.

Appendix C References

C1. Dobranich, Dean, SAFSIM Input Manual -- A Computer Program for the Engineering Simulation of Flow Systems, SAND92-0694, Sandia National Laboratories, Albuquerque, NM, September 1992.

C2. Incropera, F. P. and D. P. DeWitt, *Introduction to Heat Transfer*, John Wiley & Sons, Inc., New York, NY, 1985.

C3. Chapman, Alan J. *Fundamentals of Heat Transfer*, Macmillan Publishing Company, New York, NY, 1987.

C4. Achenbach, E. "Heat Transfer and Pressure Drop of Pebble Beds Up to High Reynolds Number," in *Proceedings of Seventh International Heat Transfer Conference*, Munchen, Federal Republic of Germany, Vol. 1, pp. 3-8, 1982.

Appendix D: Selected Mathematical Functions

In SAFSIM, the analyst is provided with a library of about 60 mathematical functions that can be used to control many of the input variables during a simulation. Most of the mathematical functions are common, such as square root, inverse, addition, multiplication, exponentiation, and absolute value. A few not so common, but useful, functions are provided here for reference.

The average fluid density (function ID# 123) for a specified region of a flow network or networks is calculated by

$$\overline{\rho} = \frac{\sum_{j=1}^{N_E} \rho_j V_j \delta_j}{\sum_{j=1}^{N_E} V_j \delta_j} \qquad \text{(D.1)}$$

where $\overline{\rho}$ is the average fluid density, N_E is the number of fluid mechanics elements for which the density is to be averaged, j is element index number, ρ_j is the fluid density for element j, V_j is the volume, and δ_j is a user-specified weighting factor. The weighting factor is useful in accounting for the relative importance of certain regions for use with the reactor dynamics feedback terms.

The average temperature (function ID# 122) for a specified region of a heat transfer structure or structures is calculated by

$$\overline{T} = \frac{\sum_{j=1}^{N_E} \rho_j V_j c_j \delta_j T_j}{\sum_{j=1}^{N_E} \rho_j V_j c_j \delta_j} \qquad \text{(D.2)}$$

where $\overline{T}$ is the average temperature, N_E is the number of structure finite elements for which the temperature is to be averaged, ρ_j is the density for element j, V_j is the volume, c_j is the specific heat, δ_j is a user-specified weighting factor, and T_j is the temperature. Again, the weighting factor is useful in accounting for the relative importance of certain regions for use with the reactor dynamics feedback terms. If all weighting factors are unity, internal energy is conserved.

A unit square wave (function ID# 152) is given by the following formula:

$$y = 1 \text{ if } \sin(x) > 0, \; y = 0 \text{ otherwise} \qquad \text{(D.3)}$$

where y is the function-controlled variable and x is the argument of the function.

A radiative heat flux boundary condition (function ID# 167) is provided by

$$q''_{1-2} = \frac{q_{1-2}}{A_1} = \frac{\sigma\left(T_1^4 - T_2^4\right)}{\frac{(1-e_1)}{e_1} + \frac{1}{F_{1-2}} + \frac{(1-e_2)A_1}{e_2 A_2}} \tag{D.4}$$

where q''_{1-2} is the heat flux from surface 1 to surface 2, A_1 and A_2 are the areas of surfaces 1 and 2, σ is the Stefan-Boltzmann constant, T is the temperature, e is the surface emissivity, and $F_{1\text{-}2}$ is the view factor from surface 1 to surface 2. Note that for a convex surface enclosed by an infinite concave surface, A_1/A_2 approaches zero, F_{1-2} equals 1, and Equation (D.4) simplifies to

$$q''_{1-2} = \frac{q_{1-2}}{A_1} = \sigma e_1\left(T_1^4 - T_2^4\right) \tag{D.5}$$

If only the heat flux from surface 1 is specified as a function-controlled variable, then the heat transfer is "one way." To enable a two-way exchange of heat,, surface 2 must also be specified as a function-controlled variable using the following equation:

$$q''_{2-1} = -q''_{1-2}\left(\frac{A_1}{A_2}\right) \tag{D.6}$$

where the minus sign indicates that the heat out of one surface equals the heat into the coupled surface and vise versa.

A conductive heat flux boundary condition (function ID# 168) is similarly provided by

$$q''_{1-2} = \frac{q_{1-2}}{A_1} = \frac{T_1 - T_2}{\frac{L_{s1}}{k_{s1}} + \left(\frac{A_1}{A_2}\right)\frac{L_{s2}}{k_{s2}} + \left(\frac{A_1}{A_c}\right)\frac{1}{h_{\text{con}}}} \tag{D.7}$$

where L_s is the conduction length associated with the surface, k_s is the thermal conductivity, A_c is the area of contact between the two surfaces, and h_{con} is an effective contact heat transfer coefficient that accounts for contact resistance. Again, Equation (D.6) should be used for two-way heat transfer.

Appendix E: Modified Heat Transfer Coefficients

Cylinders

The following equations, generalized from Reference E1, can be used to determine the modified heat transfer coefficient for lumped-capacitance models of multilayered cylindrical structures:

$$\bar{h}_c = \frac{U h_c}{\overline{w} h_c + U} \qquad \text{(E.1)}$$

$$U = \left[\sum_{j=1}^{n} \frac{1}{U_j} \right]^{-1} \qquad \text{(E.2)}$$

$$U_0 = \infty \qquad \text{(E.3)}$$

$$U_j = \frac{k_j}{R} \left[\frac{1}{\ln\left(\frac{r_j}{r_{j+1}}\right)} \right] \quad \text{for } j = 1 \text{ to } n - 1 \qquad \text{(E.4)}$$

$$U_n = \frac{2k_n}{R} \qquad \text{(E.5)}$$

$$\overline{w} = \frac{\sum_{j=1}^{n} m_j c_j \left(w_j + w_{j-1}\right)}{2 \sum_{j=1}^{n} m_j c_j} \qquad \text{(E.6)}$$

$$w_0 = 0 \qquad \text{(E.7)}$$

$$w_j = \frac{U}{U_j} + w_{j-1} \quad \text{for } j = 1 \text{ to } n - 1 \qquad \text{(E.8)}$$

$$w_n = 1 \qquad \text{(E.9)}$$

where $\overline{h_c}$ is the modified heat transfer coefficient for a multiregion cylinder, U is the effective inverse resistance for the cylinder, h_c is the unmodified heat transfer coefficient, $\overline{w}$ is an effective weighting factor for the cylinder, n is the number of radial regions, j is the region index (j = 1 for the outermost radial region), U_j is the inverse resistance of region j, k_j is the thermal conductivity of region j, R is the outside radius of the cylinder, r_j is the outside radius of region j, m_j is the mass of region j, c_j is the specific heat of region j, and w_j is a weighting factor for region j. U and $\overline{w}$ can be determined as functions of temperature if desired. The unmodified heat transfer coefficient can be determined by any correlation appropriate for the problem geometry.

Spheres

The following equations, generalized from Reference E2, can be used to determine the modified heat transfer coefficient for lumped-capacitance models of multilayered spherical structures:

$$\overline{h_c} = \frac{Uh_c}{\overline{w}h_c + U} \tag{E.10}$$

$$U = \left[\sum_{j=1}^{n} \frac{1}{U_j}\right]^{-1} \tag{E.11}$$

$$U_0 = \infty \tag{E.12}$$

$$U_j = \frac{k_j}{R^2}\left(\frac{r_j r_{j+1}}{r_j - r_{j+1}}\right) \quad \text{for } j = 1 \text{ to } n - 1 \tag{E.13}$$

$$U_n = \frac{2k_n}{3R} \tag{E.14}$$

$$\overline{w} = \frac{\sum_{j=1}^{n} m_j c_j \left(w_j + w_{j-1}\right)}{2\sum_{j=1}^{n} m_j c_j} \tag{E.15}$$

$$w_0 = 0 \tag{E.16}$$

$$w_j = \frac{U}{U_j} + w_{j-1} \quad \text{for} j = 1 \text{ to } n - 1 \tag{E.17}$$

$$w_n = 1 \tag{E.18}$$

where $\overline{h}_c$ is the modified heat transfer coefficient for a multiregion sphere, U is the effective inverse resistance for the sphere, h_c is the unmodified heat transfer coefficient, $\overline{w}$ is an effective weighting factor for the sphere, n is the number of radial regions, j is the region index (j = 1 for the outermost radial region), U_j is the inverse resistance of region j, k_j is the thermal conductivity of region j, R is the outside radius of the sphere, r_j is the outside radius of region j, m_j is the mass of region j, c_j is the specific heat of region j, and w_j is a weighting factor for region j. U and $\overline{w}$ can be determined as functions of temperature if desired. The unmodified heat transfer coefficient can be determined by any correlation appropriate for the problem geometry.

It should be noted that the temperatures predicted by such modified lumped-capacitance models are not really average temperatures. Instead, the predicted temperatures represent the effective temperatures required to provide the correct (approximately) transient heat transfer response.

Appendix E References

E1. Meyer, John E., Some Physical and Numerical Considerations for the SSC-S Code, NUREG/CR-0451, BNL-NUREG-50913, Brookhaven National Laboratory, Utpon, NY, September 1978.

E2. Tuddenham, Read Stapley, "Thermal Hydraulic Analysis of a Packed Bed Reactor Fuel Element," Masters Thesis, Massachusetts Institute of Technology, Boston, MA, May 1989.

Intentionally Blank

Appendix F: SAFSIM Sample Problem

This appendix provides the output for a sample problem that demonstrates a typical application of SAFSIM. The example problem concerns a viscosity-driven gas flow instability (sometimes called laminar flow instability) in two parallel flow channels filled with a power-generating packed bed of spherical particles. This type of instability arises because the viscosity of a gas increases with temperature which in turn increases the flow resistance. The details of the phenomena and the SAFSIM input model are explained in Reference F1. The SAFSIM output for the problem (pages 164 – 186) is presented to provide an example of a typical application. Reviewing the output files provides an introduction as to how such a problem can be modeled with SAFSIM.

Figure F1 shows a schematic of the two-channel input model along with the predicted bed surface temperature for the two channels as a function of time. One of the channels has a porosity of 0.4 and the other has a porosity of 0.38. The flow redistributes to the 0.4 porosity channel, starving the other channel of flow and resulting in a temperature excursion. The instability is initiated by decreasing the inlet mass flow rate (specified by a time-dependent boundary condition) below the critical value required for stability.

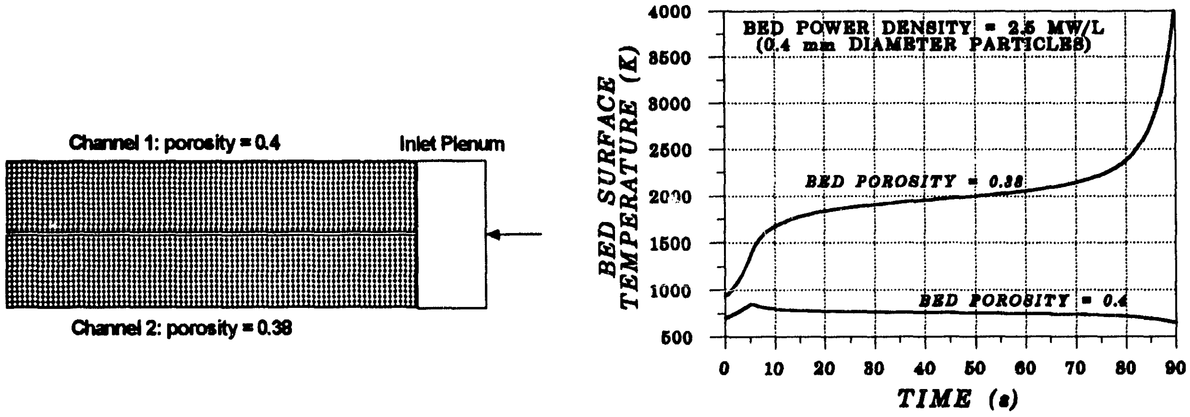


Figure F1. Temperature Excursion Due to a Flow Instability.

Appendix F References

F1. Dobranich, Dean, "A Simple Description of Flow Instability with Application to a Packed Bed," accepted for publication in the Proceedings of the 11th Symposium on Space Nuclear Power and Propulsion, Albuquerque, NM, M. S. El-Genk and M. D. Hoover, eds., January 1994.

```
  EXECUTION-TIME DATE: 09/13/93                                                      SAFSIM: COPYRIGHT 1993, SANDIA CORPORATION
---------------------------------------------------------------------------------------------------------------------------------
                                             PROGRAM SAFSIM (Version 1.2) INPUT DATA:
                                                    (ALL UNITS ARE SI/mKs)

EXECUTION DATA:

  PROBLEM TITLE --
 EXAMPLE PROBLEM - INVESTIGATE STABILITY OF FLOW IN PARALLEL FLOW PATHS
 CONTAINING POROUS BEDS WITH INTERNAL HEAT GENERATION

---------------------------------------------------------------------------------------------------------------------------------

  PROBLEM RESTART IS OFF
  MAXIMUM NUMBER OF STEADY-STATE WALL TEMPERATURE ITERATIONS = 1000
  ABSOLUTE ERROR ON WALL TEMPERATURE FOR STEADY STATE CALCULATIONS (K) =   0.1000
  RELATIVE ERROR ON WALL TEMPERATURE FOR STEADY STATE CALCULATIONS =   1.2500E-04
  RELAXATION PARAMETER FOR STEADY-STATE CALCULATIONS =   1.6800

  PROBLEM START TIME (s) =  0.000
  PROBLEM END TIME (s) =   90.0
  PROBLEM START TIME STEP NUMBER =      0
  SYSTEM TIME STEP (s) =   1.0000E-02

  SYSTEM PRINT INTERVAL (s) =   30.0
  SYSTEM PLOT FREQUENCY (steps) =   100
  SYSTEM DUMP FREQUENCY (steps) =   100
---------------------------------------------------------------------------------------------------------------------------------

  ACCELERATION DUE TO GRAVITY (m/sq.s) =     9.8066
  DEFAULT VALUE FOR ELEMENT MASS FLOW RATES (kg/s) =   2.0000E-08
  REYNOLDS NUMBER FOR TRANSITION TO TURBULANT FLOW (FOR FRICTION FACTORS) =   4.0000E+03
  COMPRESSOR FRACTIONAL CUTOFF SPEED =   0.0500
  DEFAULT NODE TEMPERATURE FOR ALL HEAT TRANSFER STRUCTURES (K) =   205.000
  DEFAULT FLUID DISPERSION CONDUCTIVITY COEFFICIENT (FOR POROUS MEDIA) =   0.0000

  ACHENBACH  POROUS MEDIA FRICTION FACTOR; USER-SUPPLIED CORRELATION # 1 (IPOR = 5):
    FRICTION FACTOR = (1.0-POR)/POR**3 *( 320.000 * ((1.0-POR)/REY)**   1.000 +  20.000 * ((1.0-POR)/REY)**   0.400 +   1.750)

  MOD ACHEN  POROUS MEDIA FRICTION FACTOR; USER-SUPPLIED CORRELATION # 2 (IPOR = 6):
    FRICTION FACTOR = (1.0-POR)/POR**3 *( 320.000 * ((1.0-POR)/REY)**   1.100 +  22.000 * ((1.0-POR)/REY)**   0.450 +   1.750)

---------------------------------------------------------------------------------------------------------------------------------

  ITERATION CONTROL PARAMETERS FOR FLOW NETWORK #    1

   MAXIMUM-ALLOWED ITERATIONS FOR PRESSURE SOLUTION =    0
   MAXIMUM-ALLOWED ITERATIONS FOR MASS FLOW RATE SOLUTION =    85
   MAXIMUM-ALLOWED ITERATIONS FOR DENSITY SOLUTION =   50
   MAXIMUM-ALLOWED ITERATIONS FOR NODAL TEMPERATURE SOLUTION =    5
   MAXIMUM-ALLOWED ITERATIONS FOR ELEMENT TEMPERATURE SOLUTION =    50
   MAXIMUM-ALLOWED ITERATIONS FOR MASS FRACTION SOLUTION =   10

   DESIRED RELATIVE ERROR FOR PRESSURE SOLUTION =   1.0000E-09
   DESIRED RELATIVE ERROR FOR MASS FLOW RATE SOLUTION =   1.0000E-05
   DESIRED RELATIVE ERROR FOR DENSITY SOLUTION =   1.0000E-03
   DESIRED RELATIVE ERROR FOR NODAL TEMPERATURE SOLUTION =   1.0000E-04
   DESIRED RELATIVE ERROR FOR ELEMENT TEMPERATURE SOLUTION =   2.5000E-03
   DESIRED RELATIVE ERROR FOR MASS FRACTION SOLUTION =   1.0000E-03

   RELAXATION PARAMETER FOR PRESSURE SOLUTION =  1.100
   RELAXATION PARAMETER FOR MASS FLOW RATE SOLUTION =  1.200
   RELAXATION PARAMETER FOR DENSITY SOLUTION =  1.000
   RELAXATION PARAMETER FOR NODAL TEMPERATURE SOLUTION =  1.000
   RELAXATION PARAMETER FOR ELEMENT TEMPERATURE SOLUTION =  1.000
   RELAXATION PARAMETER FOR MASS FRACTION SOLUTION =  1.000

   FLUID MECHANICS DESIRED TIME STEP (s) =   1.0000E-02     PRINT FREQUENCY (fluid mechanics time steps) =       10000
   IDYNFM =  0  ( 0 FOR QUASI-STATIC, 1 FOR DYNAMIC FLUID MECHANICS SOLUTION )

---------------------------------------------------------------------------------------------------------------------------------

  FLOW NETWORK #    1  NUMBER OF NODES:    26    NUMBER OF ELEMENTS:    25
                       NUMBER OF SUPER NODES:     4    NUMBER OF SUPER ELEMENTS:      3
   NETWORK NAME: PARALLEL BEDS
   THE SEMI-BANDWIDTH OF THE PRESSURE COEFFICIENT MATRIX =      2
   (THE MECHANICAL ENERGY EQUATION IS INITIALLY ON  )
   (CHOLESKY DECOMPOSITON SOLVER WILL BE USED IF GAUSS-SEIDEL ITERATION FOR PRESSURE FAILS)
---------------------------------------------------------------------------------------------------------------------------------
---------------------------------------------------------------------------------------------------------------------------------

  TOTAL NUMBER OF FLUID MECHANICS ELEMENTS =    25
  TOTAL NUMBER OF FLUID MECHANCIS NODES =    26
  TOTAL NUMBER OF FLUID MECHANICS SUPER ELEMENTS =     3
  TOTAL NUMBER OF FLUID MECHANICS SUPER NODES =     4
  TOTAL NUMBER OF FLUID MECHANICS CHOKED FLOW BOUNDARY ELEMENTS =     0
  TOTAL NUMBER OF FLUID MECHANICS COMPRESSORS ELEMENTS =     0
  TOTAL NUMBER OF FLUID MECHANICS DISTRIBUTED FLOW MANIFOLD ELEMENTS =     0
  TOTAL NUMBER OF HEAT TRANSFER STRUCTURES =    2
  TOTAL NUMBER OF HEAT TRANSFER STRUCTURE ELEMENTS =     20
  TOTAL NUMBER OF NUCLEAR REACTORS =    0

---------------------------------------------------------------------------------------------------------------------------------
```

```
-----------------------------------------------------------------------------------------------------------------------

GAS/LIQUID EQUATION OF STATE AND PROPERTY INPUT DATA:

  FLOW NETWORK #    1 CONTAINS  1 FLUID CONSTITUENT(s):  H2CONSTANT

-----------------------------------------------------------------------------------------------------------------------


 * GAS ID #  1 IS HYDROGEN    , THE GAS CONSTANT IS  4.1242E+03 J/kg*K  (MOLECULAR WEIGHT:   2.0160E+00)
    DENSITY (kg/cub.m) = PRESSURE/(Rgas*TEMPERATURE)

 VISCOSITY TEMPERATURE RANGE #  1: T =    50.000 TO   3300.000 K
    VISCOSITY IS DETERMINED USING THE POWER-     LAW EQUATION
    VISCOSITY (Pa*s) =  1.000E+00 *( 8.4110E-06*(T /  2.7311E+02)** 6.8000E-01)
 SPECIFIC HEAT TEMPERATURE RANGE #  1: T =   300.000 TO   1500.000 K
    CONSTANT PRESSURE SPECIFIC HEAT (J/kg*K) =  1.000E+00 *( 1.6274E+04 + -6.2496E+00*T +  6.8411E-03*T**2 + -1.8753E-06*T**3)
 SPECIFIC HEAT TEMPERATURE RANGE #  2: T =  1500.000 TO   3300.000 K
    CONSTANT PRESSURE SPECIFIC HEAT (J/kg*K) =  1.000E+00 *(-6.2935E+03 +  3.4410E+01*T + -1.8140E-02*T**2 +  3.3939E-06*T**3)
 CONDUCTIVITY TEMPERATURE RANGE #  1: T =    50.000 TO   3300.000 K
    CONDUCTIVITY (W/m*K) =  1.000E+00 *( 6.2265E-02 +  6.8021E-04*T + -4.3262E-07*T**2 +  1.3360E-10*T**3)


 * GAS ID #  2 IS H2CONSTANT  , THE GAS CONSTANT IS  4.1242E+03 J/kg*K  (MOLECULAR WEIGHT:   2.0160E+00)
    DENSITY (kg/cub.m) = PRESSURE/(Rgas*TEMPERATURE)

 VISCOSITY TEMPERATURE RANGE #  1: T =    50.000 TO   5000.000 K
    VISCOSITY IS DETERMINED USING THE POWER-     LAW EQUATION
    VISCOSITY (Pa*s) =  1.000E+00 *( 2.0000E-07*(T /  1.0000E+00)** 6.7778E-01)
 SPECIFIC HEAT TEMPERATURE RANGE #  1: T =    50.000 TO   3000.000 K
    CONSTANT PRESSURE SPECIFIC HEAT (J/kg*K) =  1.000E+00 *( 1.5000E+04 +  0.0000E+00*T +  0.0000E+00*T**2 +  0.0000E+00*T**3)
 CONDUCTIVITY TEMPERATURE RANGE #  1: T =    50.000 TO   3000.000 K
    CONDUCTIVITY (W/m*K) =  1.000E+00 *( 2.0000E-01 +  0.0000E+00*T +  0.0000E+00*T**2 +  0.0000E+00*T**3)


 * GAS ID #  3 IS HELIUM      , THE GAS CONSTANT IS  2.0730E+03 J/kg*K  (MOLECULAR WEIGHT:   4.0108E+00)
    DENSITY (kg/cub.m) = PRESSURE/(Rgas*TEMPERATURE)

 VISCOSITY TEMPERATURE RANGE #  1: T =   130.000 TO   2000.000 K
    VISCOSITY IS DETERMINED USING THE POWER-     LAW EQUATION
    VISCOSITY (Pa*s) =  1.000E+00 *( 1.8700E-05*(T /  2.7311E+02)** 6.6600E-01)
 SPECIFIC HEAT TEMPERATURE RANGE #  1: T =    50.000 TO   2000.000 K
    CONSTANT PRESSURE SPECIFIC HEAT (J/kg*K) =  1.000E+00 *( 5.1926E+03 +  0.0000E+00*T +  0.0000E+00*T**2 +  0.0000E+00*T**3)
 CONDUCTIVITY TEMPERATURE RANGE #  1: T =    50.000 TO   2000.000 K
    CONDUCTIVITY (W/m*K) =  1.000E+00 *( 3.7220E-02 +  3.8960E-04*T + -7.4500E-08*T**2 +  1.2900E-11*T**3)

-----------------------------------------------------------------------------------------------------------------------
```

```
--------------------------------------------------------------------------------------------------------

ELEMENT AND NODE INPUT DATA:

*** FLOW NETWORK #  1      NETWORK NAME: PARALLEL BEDS              ***

   E#  NODE1  NODE2  LENGTH (m)   DIAMETER (m)  Z2 - Z1 (m)    POROSITY   IPOR     NAME
   1     1      2   1.0000E-03    1.4140E-01   0.0000E+00    1.0000      0    INLET
   2     2      3   1.0000E-03    1.4140E-01   0.0000E+00    1.0000      0    INLET
   3     3      4   1.0000E-03    1.4140E-01   0.0000E+00    1.0000      0    INLET
   4     4      5   1.0000E-03    1.4140E-01   0.0000E+00    1.0000      0    INLET
   5     5      6   1.0000E-03    1.4140E-01   0.0000E+00    1.0000      0    INLET
 :
   6     6      7   1.0000E-03    4.0000E-04   0.0000E+00    0.4000      2    BED1
   7     7      8   1.0000E-03    4.0000E-04   0.0000E+00    0.4000      2    BED1
   8     8      9   1.0000E-03    4.0000E-04   0.0000E+00    0.4000      2    BED1
   9     9     10   1.0000E-03    4.0000E-04   0.0000E+00    0.4000      2    BED1
  10    10     11   1.0000E-03    4.0000E-04   0.0000E+00    0.4000      2    BED1
 :
  11    11     12   1.0000E-03    4.0000E-04   0.C000E+00    0.4000      2    BED1
  12    12     13   1.0000E-03    4.0000E-04   0.0000E+00    0.4000      2    BED1
  13    13     14   1.0000E-03    4.0000E-04   0.0000E+00    0.4000      2    BED1
  14    14     15   1.0000E-03    4.0000E-04   0.0000E+00    0.4000      2    BED1
  15    15     16   1.0000E-03    4.0000E-04   0.0000E+00    0.4000      2    BED1
 :
  16     6     17   1.0000E-03    4.0000E-04   0.0000E+00    0.3800      2    BED2
  17    17     18   1.0000E-03    4.0000E-04   0.0000E+00    0.3800      2    BED2
  18    18     19   1.0000E-03    4.0000E-04   0.0000E+00    0.3800      2    BED2
  19    19     20   1.0000E-03    4.0000E-04   0.0000E+00    0.3800      2    BED2
  20    20     21   1.0000E-03    4.0000E-04   0.0000E+00    0.3800      2    BED2
 :
  21    21     22   1.0000E-03    4.0000E-04   0.0000E+00    0.3800      2    BED2
  22    22     23   1.0000E-03    4.0000E-04   0.0000E+00    0.3800      2    BED2
  23    23     24   1.0000E-03    4.0000E-04   0.0000E+00    0.3800      2    BED2
  24    24     25   1.0000E-03    4.0000E-04   0.0000E+00    0.3800      2    BED2
  25    25     26   1.0000E-03    4.0000E-04   0.0000E+00    0.3800      2    BED2
 :
  E#  NODE1  NODE2  LENGTH (m)   DIAMETER (m)  Z2 - Z1 (m)    POROSITY   IPOR     NAME
--------------------------------------------------------------------------------------------------------
  E#  NODE1  NODE2  AREA1 (sq.m) AREA2 (sq.m) AREA (sq.m)  VOLUME (cub.m)  WALL AREA (sq.m)    NAME
   1     1      2    2.0000E-04   2.0000E-04   2.0000E-04    2.0000E-07       1.0000E-16     INLET
   2     2      3    2.0000E-04   2.0000E-04   2.0000E-04    2.0000E-07       1.0000E-16     INLET
   3     3      4    2.0000E-04   2.0000E-04   2.0000E-04    2.0000E-07       1.0000E-16     INLET
   4     4      5    2.0000E-04   2.0000E-04   2.0000E-04    2.0000E-07       1.0000E-16     INLET
   5     5      6    2.0000E-04   2.0000E-04   2.0000E-04    2.0000E-07       1.0000E-16     INLET
 :
   6     6      7    1.0000E-04   1.0000E-04   1.0000E-04    4.0000E-08       9.0000E-04     BED1
   7     7      8    1.0000E-04   1.0000E-04   1.0000E-04    4.0000E-08       9.0000E-04     BED1
   8     8      9    1.0000E-04   1.0000E-04   1.0000E-04    4.0000E-08       9.0000E-04     BED1
   9     9     10    1.0000E-04   1.0000E-04   1.0000E-04    4.0000E-08       9.0000E-04     BED1
  10    10     11    1.0000E-04   1.0000E-04   1.0000E-04    4.0000E-08       9.0000E-04     BED1
 :
  11    11     12    1.0000E-04   1.0000E-04   1.0000E-04    4.0000E-08       9.0000E-04     BED1
  12    12     13    1.0000E-04   1.0000E-04   1.0000E-04    4.0000E-08       9.0000E-04     BED1
  13    13     14    1.0000E-04   1.0000E-04   1.0000E-04    4.0000E-08       9.0000E-04     BED1
  14    14     15    1.0000E-04   1.0000E-04   1.0000E-04    4.0000E-08       9.0000E-04     BED1
  15    15     16    1.0000E-04   1.0000E-04   1.0000E-04    4.0000E-08       9.0000E-04     BED1
 :
  16     6     17    1.0000E-04   1.0000E-04   1.0000E-04    3.8000E-08       9.0000E-04     BED2
  17    17     18    1.0000E-04   1.0000E-04   1.0000E-04    3.8000E-08       9.0000E-04     BED2
  18    18     19    1.0000E-04   1.0000E-04   1.0000E-04    3.8000E-08       9.0000E-04     BED2
  19    19     20    1.0000E-04   1.0000E-04   1.0000E-04    3.8000E-08       9.0000E-04     BED2
  20    20     21    1.0000E-04   1.0000E-04   1.0000E-04    3.8000E-08       9.0000E-04     BED2
 :
  21    21     22    1.0000E-04   1.0000E-04   1.0000E-04    3.8000E-08       9.0000E-04     BED2
  22    22     23    1.0000E-04   1.0000E-04   1.0000E-04    3.8000E-08       9.0000E-04     BED2
  23    23     24    1.0000E-04   1.0000E-04   1.0000E-04    3.8000E-08       9.0000E-04     BED2
  24    24     25    1.0000E-04   1.0000E-04   1.0000E-04    3.8000E-08       9.0000E-04     BED2
  25    25     26    1.0000E-04   1.0000E-04   1.0000E-04    3.8000E-08       9.0000E-04     BED2
 :
  E#  NODE1  NODE2  AREA1 (sq.m) AREA2 (sq.m) AREA (sq.m)  VOLUME (cub.m)  WALL AREA (sq.m)    NAME
--------------------------------------------------------------------------------------------------------
                        TOTAL VOLUME IN NETWORK #   1 =  1.7800E-06 cub.m
--------------------------------------------------------------------------------------------------------
  E#  NODE1  NODE2  RELRUF/DKC      AK12         AK21         ALOD         NAME
   1     1      2   0.0000E+00   0.0000E+00   0.0000E+00   0.0000E+00   INLET
   2     2      3   0.0000E+00   0.0000E+00   0.0000E+00   0.0000E+00   INLET
   3     3      4   0.0000E+00   0.0000E+00   0.0000E+00   0.0000E+00   INLET
   4     4      5   0.0000E+00   0.0000E+00   0.0000E+00   0.0000E+00   INLET
   5     5      6   0.0000E+00   4.0000E-01   4.0000E-01   0.0000E+00   INLET
 :
   6     6      7   0.0000E+00   0.0000E+00   0.0000E+00   0.0000E+00   BED1
   7     7      8   0.0000E+00   0.0000E+00   0.0000E+00   0.0000E+00   BED1
   8     8      9   0.0000E+00   0.0000E+00   0.0000E+00   0.0000E+00   BED1
   9     9     10   0.0000E+00   0.0000E+00   0.0000E+00   0.0000E+00   BED1
  10    10     11   0.0000E+00   0.0000E+00   0.0000E+00   0.0000E+00   BED1
 :
  11    11     12   0.0000E+00   0.0000E+00   0.0000E+00   0.0000E+00   BED1
  12    12     13   0.0000E+00   0.0000E+00   0.0000E+00   0.0000E+00   BED1
  13    13     14   0.0000E+00   0.0000E+00   0.0000E+00   0.0000E+00   BED1
  14    14     15   0.0000E+00   0.0000E+00   0.0000E+00   0.0000E+00   BED1
  15    15     16   0.0000E+00   0.0000E+00   0.0000E+00   0.0000E+00   BED1
 :
 16     6     17   0.0000E+00   0.0000E+00   0.0000E+00   0.0000E+00    BED2
 17    17     18   0.0000E+00   0.0000E+00   0.0000E+00   0.0000E+00    BED2
 18    18     19   0.0000E+00   0.0000E+00   0.0000E+00   0.0000E+00    BED2
 19    19     20   0.0000E+00   0.0000E+00   0.0000E+00   0.0000E+00    BED2
 20    20     21   0.0000E+00   0.0000E+00   0.0000E+00   0.0000E+00    BED2
 :
 21    21     22   0.0000E+00   0.0000E+00   0.0000E+00   0.0000E+00    BED2
 22    22     23   0.0000E+00   0.0000E+00   0.0000E+00   0.0000E+00    BED2
 23    23     24   0.0000E+00   0.0000E+00   0.0000E+00   0.0000E+00    BED2
 24    24     25   0.0000E+00   0.0000E+00   0.0000E+00   0.0000E+00    BED2
 25    25     26   0.0000E+00   0.0000E+00   0.0000E+00   0.0000E+00    BED2
 :
 E#  NODE1  NODE2  RELRUF/DKC      AK12         AK21         ALOD           NAME
------------------------------------------------------------------------------------------------------------------------
 E#  NODE1  NODE2    P1 (Pa)      P2 (Pa)      P (Pa)      T1 (K)      T2 (K)      T (K)      FLOW (kg/s)  QDIRECT (W)
  1     1      2   7.6000E+06   7.6000E+06   7.6000E+06   125.000     125.000     125.000    2.0000E-08   0.0000E+00
  2     2      3   7.6000E+06   7.6000E+06   7.6000E+06   125.000     125.000     125.000    2.0000E-08   0.0000E+00
  3     3      4   7.6000E+06   7.6000E+06   7.6000E+06   125.000     125.000     125.000    2.0000E-08   0.0000E+00
  4     4      5   7.6000E+06   7.6000E+06   7.6000E+06   125.000     125.000     125.000    2.0000E-08   0.0000E+00
  5     5      6   7.6000E+06   7.6000E+06   7.6000E+06   125.000     125.000     125.000    2.0000E-08   0.0000E+00
 :
  6     6      7   7.6000E+06   7.6000E+06   7.6000E+06   125.000     200.000     162.500    2.0000E-08   0.0000E+00
  7     7      8   7.6000E+06   7.6000E+06   7.6000E+06   200.000     200.000     200.000    2.0000E-08   0.0000E+00
  8     8      9   7.6000E+06   7.6000E+06   7.6000E+06   200.000     200.000     200.000    2.0000E-08   0.0000E+00
  9     9     10   7.6000E+06   7.6000E+06   7.6000E+06   200.000     200.000     200.000    2.0000E-08   0.0000E+00
 10    10     11   7.6000E+06   7.6000E+06   7.6000E+06   200.000     200.000     200.000    2.0000E-08   0.0000E+00
 :
 11    11     12   7.6000E+06   7.6000E+06   7.6000E+06   200.000     200.000     200.000    2.0000E-08   0.0000E+00
 12    12     13   7.6000E+06   7.6000E+06   7.6000E+06   200.000     200.000     200.000    2.0000E-08   0.0000E+00
 13    13     14   7.6000E+06   7.6000E+06   7.6000E+06   200.000     200.000     200.000    2.0000E-08   0.0000E+00
 14    14     15   7.6000E+06   7.6000E+06   7.6000E+06   200.000     200.000     200.000    2.0000E-08   0.0000E+00
 15    15     16   7.6000E+06   7.5778E+06   7.5889E+06   200.000     200.000     200.000    2.0000E-08   0.0000E+00
 :
 16     6     17   7.6000E+06   7.6000E+06   7.6000E+06   125.000     200.000     162.500    2.0000E-08   0.0000E+00
 17    17     18   7.6000E+06   7.6000E+06   7.6000E+06   200.000     200.000     200.000    2.0000E-08   0.0000E+00
 18    18     19   7.6000E+06   7.6000E+06   7.6000E+06   200.000     200.000     200.000    2.0000E-08   0.0000E+00
 19    19     20   7.6000E+06   7.6000E+06   7.6000E+06   200.000     200.000     200.000    2.0000E-08   0.0000E+00
 20    20     21   7.6000E+06   7.6000E+06   7.6000E+06   200.000     200.000     200.000    2.0000E-08   0.0000E+00
 :
 21    21     22   7.6000E+06   7.6000E+06   7.6000E+06   200.000     200.000     200.000    2.0000E-08   0.0000E+00
 22    22     23   7.6000E+06   7.6000E+06   7.6000E+06   200.000     200.000     200.000    2.0000E-08   0.0000E+00
 23    23     24   7.6000E+06   7.6000E+06   7.6000E+06   200.000     200.000     200.000    2.0000E-08   0.0000E+00
 24    24     25   7.6000E+06   7.6000E+06   7.6000E+06   200.000     200.000     200.000    2.0000E-08   0.0000E+00
 25    25     26   7.6000E+06   7.5778E+06   7.5889E+06   200.000     200.000     200.000    2.0000E-08   0.0000E+00
 :
 E#  NODE1  NODE2    P1 (Pa)      P2 (Pa)      P (Pa)      T1 (K)      T2 (K)      T (K)      FLOW (kg/s)  QDIRECT (W)

------------------------------------------------------------------------------------------------------------------------
SUPER ELEMENT AND SUPER NODE DATA:

  SE#:    1     NODE1S:     1     NODE2S:     2     NUMBER OF ELEMENTS:    5
  ELEMENT NUMBERS:
    1     2     3     4     5
  SE#:    2     NODE1S:     2     NODE2S:     3     NUMBER OF ELEMENTS:   10
  ELEMENT NUMBERS:
    6     7     8     9    10    11    12    13    14    15
  SE#:    3     NODE1S:     2     NODE2S:     4     NUMBER OF ELEMENTS:   10
  ELEMENT NUMBERS:
   16    17    18    19    20    21    22    23    24    25
------------------------------------------------------------------------------------------------------------------------
------------------------------------------------------------------------------------------------------------------------
```

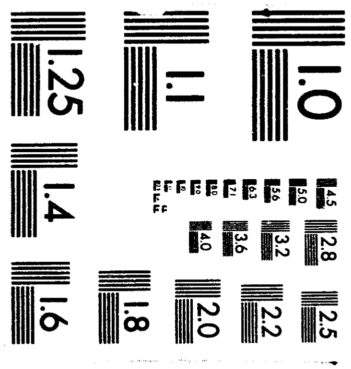
```
------------------------------------------------------------------------------------------------------------------------------

BOUNDARY CONDITION IDENTIFICATION INPUT DATA:

*** FLOW NETWORK #  1      NETWORK NAME: PARALLEL BEDS                ***

    BC KIND           SUPER NODE NUMBER      NODE NUMBER      SET ID #

  MASS FLOW RATE              1                   1              101
  PRESSURE                    3                  16              102
  PRESSURE                    4                  26              102
------------------------------------------------------------------------------------------------------------------------------

BOUNDARY CONDITION SET INPUT DATA:

*** FLOW NETWORK #  1      NETWORK NAME: PARALLEL BEDS                ***

  SET ID #        BC VALUE       TEMPERATURE BC     MASS FRACTION BC(S)

     101        1.0000E-03          125.000          1.0000
     102        7.5778E+06         -300.000          1.0000
------------------------------------------------------------------------------------------------------------------------------
```

```
--------------------------------------------------------------------------------------------------------------------

FUNCTION-CONTROLLED VARIABLE INPUT DATA FOR FLUID MECHANICS:

     COUNTER   ID #    ............NAME............    ARGUMENT F #     MULTIPLIER    L1    L2    L3

--------------------------------------------------------------------------------------------------------------------


FUNCTION INPUT DATA FOR FLUID MECHANICS:

     COUNTER   F #    ID #     ............NAME............    MULTIPLIER       VALUE       L1    L2    L3    L4

         1       1       8    P at BED INLET                   1.0000E+00     8.8153E+05     1      6     0     0
         2       2       8    P at BED1 OUTLET                 1.0000E+00     8.8153E+05     1     16     0     0
         3       3     132    Pin - Pout OF BED1               1.0000E+00     0.0000E+00     0      0     0     0
                              ASSOCIATED ARGUMENT F #s:
                                       1      2      0      0      0      0
         4       6       9    Tgas at BED1 INLET               1.0000E+00     1.2500E+02     1      6     0     0
         5       7       9    Tgas at BED1 OUTLET              1.0000E+00     1.2500E+02     1     16     0     0
         6       8     132    Tout - Tin OF BED1               1.0000E+00     0.0000E+00     0      0     0     0
                              ASSOCIATED ARGUMENT F #s:
                                       7      6      0      0      0      0
         7       9     134    phi=(Tout-Tin)/Tin OF BED1       1.0000E+00     0.0000E+00     0      0     0     0
                              ASSOCIATED ARGUMENT F #s:
                                       8      6      0      0      0      0
         8      17       9    Tgas at BED2 OUTLET              1.0000E+00     1.2500E+02     1     26     0     0
         9      18     132    Tout - Tin OF BED2               1.0000E+00     0.0000E+00     0      0     0     0
                              ASSOCIATED ARGUMENT F #s:
                                      17      6      0      0      0      0
        10      19     134    phi=(Tout-Tin)/Tin OF BED2       1.0000E+00     0.0000E+00     0      0     0     0
                              ASSOCIATED ARGUMENT F #s:
                                      18      6      0      0      0      0
--------------------------------------------------------------------------------------------------------------------
```

```
------------------------------------------------------------------------------------------------------------------------

HEAT TRANSFER STRUCTURE MATERIAL PROPERTY INPUT DATA:


   * MATERIAL #   1      MATERIAL NAME: BED,D=400      MATERIAL DENSITY (kg/cub.m) =   5.9300E+03

   SPECIFIC HEAT DATA:
      TABLE TYPE  1   WITH   6  DATA PAIRS,      MULTIPLIER =   1.0000E+00
            TEMPERATURE (K)    SPECIFIC HEAT (J/kg*K)
              3.0000E+02          3.5300E+02
              3.9000E+02          4.3800E+02
              4.9300E+02          4.6700E+02
              1.0730E+03          5.1600E+02
              1.6000E+03          5.3200E+02
              1.9550E+03          5.3800E+02


   CONDUCTIVITY DATA:
      POWER LAW,      MULTIPLIER =   1.0000E+00
            CONDUCTIVITY (W/m*K) =   1.0000E-10 * [T (K)]**  3.0000E+00 +  1.0000E-02

   * MATERIAL #   2      MATERIAL NAME: BED,D=400C     MATERIAL DENSITY (kg/cub.m) =   3.5580E+03

   SPECIFIC HEAT DATA:
      CONSTANT,      MULTIPLIER =   1.0000E+00
            SPECIFIC HEAT (J/kg/K) =   3.6000E+02

   CONDUCTIVITY DATA:
      CONSTANT,      MULTIPLIER =   1.0000E+00
            CONDUCTIVITY (W/m*K) =   2.0000E+00

   * MATERIAL #   3      MATERIAL NAME: SS,POR=0.35    MATERIAL DENSITY (kg/cub.m) =   5.1350E+03

   SPECIFIC HEAT DATA:
      TABLE TYPE  1   WITH   9  DATA PAIRS,      MULTIPLIER =   1.0000E+00
            TEMPERATURE (K)    SPECIFIC HEAT (J/kg*K)
              1.0000E+02          2.7200E+02
              2.0000E+02          4.0200E+02
              3.0000E+02          4.7700E+02
              4.0000E+02          5.1500E+02
              6.0000E+02          5.5700E+02
              8.0000E+02          5.8200E+02
              1.0000E+03          6.1100E+02
              1.2000E+03          6.4000E+02
              1.5000E+03          6.8200E+02


   CONDUCTIVITY DATA:
      POLYNOMIAL OF ORDER   2,      MULTIPLIER =   6.5000E-01
            POLYNOMIAL CONSTANTS A0, A1,...,AN:
                          8.0007E+00   2.2119E-02  -4.3623E-06

   * MATERIAL #   4      MATERIAL NAME: C-C,RADIAL     MATERIAL DENSITY (kg/cub.m) =   1.2000E+03

   SPECIFIC HEAT DATA:
      CONSTANT,      MULTIPLIER =   1.0000E+00
            SPECIFIC HEAT (J/kg/K) =   8.3740E+02

   CONDUCTIVITY DATA:
      TABLE TYPE  1   WITH   6  DATA PAIRS,      MULTIPLIER =   1.0000E+00
            TEMPERATURE (K)    CONDUCTIVITY (W/m*K)
              2.2000E+02          7.7900E+01
              2.9400E+02          6.9200E+01
              5.8860E+02          5.1900E+01
              8.1080E+02          4.3300E+01
              1.3663E+03          3.4600L+01
              1.9220E+03          2.4200E+01


   * MATERIAL #   5      MATERIAL NAME: GRAPHITE       MATERIAL DENSITY (kg/cub.m) =   1.9000E+03

   SPECIFIC HEAT DATA:
      CONSTANT,      MULTIPLIER =   1.0000E+00
            SPECIFIC HEAT (J/kg/K) =   1.9674E+03

   CONDUCTIVITY DATA:
      CONSTANT,      MULTIPLIER =   1.0000E+00
            CONDUCTIVITY (W/m*K) =   3.0000E+01
------------------------------------------------------------------------------------------------------------------------
```

```
-----------------------------------------------------------------------------------------------------------------------

HEAT TRANSFER STRUCTURE ELEMENT AND NODE INPUT DATA:

   TOTAL NUMBER OF HEAT TRANSFER STRUCTURES:   2

-----------------------------------------------------------------------------------------------------------------------

   * STRUCTURE #     1      STRUCTURE NAME: BED1
     NUMBER OF COPIES =    1.0000E+00      Q''' MULTIPLIER =    1.0000E+00       KHDYN =  1
     NUMBER OF ELEMENTS =  10    MINIMUM TIME STEP (s) =    1.0000E-03   FRACTION OF EXPLICIT TIME STEP LIMIT =  0.50000
     PRINT FREQUENCY (heat transfer time steps) =   10000

   RECTANGULAR GEOMETRY WAS SELECTED:
     WIDTH (m) =    1.0000E-02    HEIGHT (m) =   1.0000E-02
     X1 (m) =    0.0000E+00    X2 (m) =    1.0000E-02

   HEAT FLUX BOUNDARY CONDITION AT END 1
     HEAT FLUX (W/sq.m) =    0.0000E+00   SURFACE AREA (sq.m) =    6.0000E+03

-----------------------------------------------------------------------------------------------------------------------
e/n#      Xn (m)       Xe (m)      DX (m)    AREA (sq.m) VOLUME (cub.m)  MASS (kg)   MATERIAL   Q''' (W/cub.m)   Te (K)      Tn (K)
  1  0.0000E+00  5.0000E-04  1.0000E-03  1.0000E-04    1.0000E-07    3.558E-04  BED,D=400C    2.5000E+09    205.000     205.000
  2  1.0000E-03  1.5000E-03  1.0000E-03  1.0000E-04    1.0000E-07    3.558E-04  BED,D=400C    2.5000E+09    205.000     205.000
  3  2.0000E-03  2.5000E-03  1.0000E-03  1.0000E-04    1.0000E-07    3.558E-04  BED,D=400C    2.5000E+09    205.000     205.000
  4  3.0000E-03  3.5000E-03  1.0000E-03  1.0000E-04    1.0000E-07    3.558E-04  BED,D=400C    2.5000E+09    205.000     205.000
  5  4.0000E-03  4.5000E-03  1.0000E-03  1.0000E-04    1.0000E-07    3.558E-04  BED,D=400C    2.5000E+09    205.000     205.000
 :
  6  5.0000E-03  5.5000E-03  1.0000E-03  1.0000E-04    1.0000E-07    3.558E-04  BED,D=400C    2.5000E+09    205.000     205.000
  7  6.0000E-03  6.5000E-03  1.0000E-03  1.0000E-04    1.0000E-07    3.558E-04  BED,D=400C    2.5000E+09    205.000     205.000
  8  7.0000E-03  7.5000E-03  1.0000E-03  1.0000E-04    1.0000E-07    3.558E-04  BED,D=400C    2.5000E+09    205.000     205.000
  9  8.0000E-03  8.5000E-03  1.0000E-03  1.0000E-04    1.0000E-07    3.558E-04  BED,D=400C    2.5000E+09    205.000     205.000
 10  9.0000E-03  9.5000E-03  1.0000E-03  1.0000E-04    1.0000E-07    3.558E-04  BED,D=400C    2.5000E+09    205.000     205.000
 :
 11  1.0000E-02                                                                                                         205.000
STRUCTURE TOTALS:   LENGTH (m) =   1.0000E-02    VOLUME (cub.m) =   1.0000E-06    MASS (kg) =   3.5580E-03
-----------------------------------------------------------------------------------------------------------------------
SPECIFIED ELEMENT HEAT FLUX SURFACES:
 e#   SURFACE   AREA (sq.m)  HEAT FLUX (W/sq.m)
-----------------------------------------------------------------------------------------------------------------------
SPECIFIED ELEMENT CONVECTION SURFACES:
 e#    SURFACE  NW #   E#    AREA (sq.m)       EDh (m)      Lam.HTC #  Tur.HTC #     REtran        Lam.AUX       Tur.AUX
  1       1       1    15    9.0000E-04   4.0000E-04       36         36       4.0000E+03    1.0000E+00    1.0000E+00
  2       1       1    14    9.0000E-04   4.0000E-04       36         36       4.0000E+03    1.0000E+00    1.0000E+00
  3       1       1    13    9.0000E-04   4.0000E-04       36         36       4.0000E+03    1.0000E+00    1.0000E+00
  4       1       1    12    9.0000E-04   4.0000E-04       36         36       4.0000E+03    1.0000E+00    1.0000E+00
  5       1       1    11    9.0000E-04   4.0000E-04       36         36       4.0000E+03    1.0000E+00    1.0000E+00
 :
  6       1       1    10    9.0000E-04   4.0000E-04       36         36       4.0000E+03    1.0000E+00    1.0000E+00
  7       1       1     9    9.0000E-04   4.0000E-04       36         36       4.0000E+03    1.0000E+00    1.0000E+00
  8       1       1     8    9.0000E-04   4.0000E-04       36         36       4.0000E+03    1.0000E+00    1.0000E+00
  9       1       1     7    9.0000E-04   4.0000E-04       36         36       4.0000E+03    1.0000E+00    1.0000E+00
 10       1       1     6    9.0000E-04   4.0000E-04       36         36       4.0000E+03    1.0000E+00    1.0000E+00
 :
-----------------------------------------------------------------------------------------------------------------------

   HEAT FLUX BOUNDARY CONDITION AT END 2
     HEAT FLUX (W/sq.m) =    0.0000E+00   SURFACE AREA (sq.m) =    6.0000E+03

-----------------------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------------------

   * STRUCTURE #     2      STRUCTURE NAME: BED2
     NUMBER OF COPIES =    1.0000E+00      Q''' MULTIPLIER =    1.0000E+00       KHDYN =  1
     NUMBER OF ELEMENTS =  10    MINIMUM TIME STEP (s) =    1.0000E-03   FRACTION OF EXPLICIT TIME STEP LIMIT =  0.50000
     PRINT FREQUENCY (heat transfer time steps) =   10000

   RECTANGULAR GEOMETRY WAS SELECTED:
     WIDTH (m) =    1.0000E-02    HEIGHT (m) =   1.0000E-02
     X1 (m) =    0.0000E+00    X2 (m) =    1.0000E-02

   HEAT FLUX BOUNDARY CONDITION AT END 1
     HEAT FLUX (W/sq.m) =    0.0000E+00   SURFACE AREA (sq.m) =    6.0000E+03

-----------------------------------------------------------------------------------------------------------------------
e/n#      Xn (m)       Xe (m)      DX (m)    AREA (sq.m) VOLUME (cub.m)  MASS (kg)   MATERIAL   Q''' (W/cub.m)   Te (K)      Tn (K)
  1  0.0000E+00  5.0000E-04  1.0000E-03  1.0000E-04    1.0000E-07    3.558E-04  BED,D=400C    2.5000E+09    205.000     205.000
  2  1.0000E-03  1.5000E-03  1.0000E-03  1.0000E-04    1.0000E-07    3.558E-04  BED,D=400C    2.5000E+09    205.000     205.000
  3  2.0000E-03  2.5000E-03  1.0000E-03  1.0000E-04    1.0000E-07    3.558E-04  BED,D=400C    2.5000E+09    205.000     205.000
  4  3.0000E-03  3.5000E-03  1.0000E-03  1.0000E-04    1.0000E-07    3.558E-04  BED,D=400C    2.5000E+09    205.000     205.000
  5  4.0000E-03  4.5000E-03  1.0000E-03  1.0000E-04    1.0000E-07    3.558E-04  BED,D=400C    2.5000E+09    205.000     205.000
 :
  6  5.0000E-03  5.5000E-03  1.0000E-03  1.0000E-04    1.0000E-07    3.558E-04  BED,D=400C    2.5000E+09    205.000     205.000
  7  6.0000E-03  6.5000E-03  1.0000E-03  1.0000E-04    1.0000E-07    3.558E-04  BED,D=400C    2.5000E+09    205.000     205.000
  8  7.0000E-03  7.5000E-03  1.0000E-03  1.0000E-04    1.0000E-07    3.558E-04  BED,D=400C    2.5000E+09    205.000     205.000
  9  8.0000E-03  8.5000E-03  1.0000E-03  1.0000E-04    1.0000E-07    3.558E-04  BED,D=400C    2.5000E+09    205.000     205.000
 10  9.0000E-03  9.5000E-03  1.0000E-03  1.0000E-04    1.0000E-07    3.558E-04  BED,D=400C    2.5000E+09    205.000     205.000
 :
 11  1.0000E-02                                                                                                         205.000
STRUCTURE TOTALS:   LENGTH (m) =   1.0000E-02    VOLUME (cub.m) =   1.0000E-06    MASS (kg) =   3.5580E-03
-----------------------------------------------------------------------------------------------------------------------
SPECIFIED ELEMENT HEAT FLUX SURFACES:
 e#   SURFACE   AREA (sq.m)  HEAT FLUX (W/sq.m)
```

```
------------------------------------------------------------------------------------------------------------------------------
  SPECIFIED ELEMENT CONVECTION SURFACES:
   e#   SURFACE  NW #   E#   AREA (sq.m)     FDh (m)     Lam.HTC #  Tur.HTC #     REtran        Lam.AUX       Tur.AUX
    1      1       1    25    9.0000E-04   4.0000E-04       36         36       4.0000E+03    1.0000E+00    1.0000E+00
    2      1       1    24    9.0000E-04   4.0C00E-04       36         36       4.0000E+03    1.0000E+00    1.0000E+00
    3      1       1    23    9.0000E-04   4.0000E-04       36         36       4.0000E+03    1.0000E+00    1.0000E+00
    4      1       1    22    9.0000E-04   4.0000E-04       36         36       4.0000E+03    1.0000E+00    1.0000E+00
    5      1       1    21    9.0000E-04   4.0000E-04       36         36       4.0000E+03    1.0000E+00    1.0000E+00
  :
    6      1       1    20    9.0000E-04   4.0000E-04       36         36       4.0000E+03    1.0000E+00    1.0000E+00
    7      1       1    19    9.0000E-04   4.0000E-04       36         36       4.0000E+03    1.0000E+00    1.0000E+00
    8      1       1    18    9.0000E-04   4.0000E-04       36         36       4.0000E+03    1.0000E+00    1.0000E+00
    9      1       1    17    9.0000E-04   4.0000E-04       36         36       4.0000E+03    1.0000E+00    1.0000E+00
   10      1       1    16    9.0000E-04   4.0000E-04       36         36       4.0000E+03    1.0000E+00    1.0000E+00
  :
------------------------------------------------------------------------------------------------------------------------------

   HEAT FLUX BOUNDARY CONDITION AT END 2
     HEAT FLUX (W/sq.m) =   0.0000E+00   SURFACE AREA (sq.m) =   6.0000E+03

------------------------------------------------------------------------------------------------------------------------------
------------------------------------------------------------------------------------------------------------------------------
  *TOTAL MASS OF ALL COPIES OF ALL STRUCTURES (kg) =   7.1160E-03
------------------------------------------------------------------------------------------------------------------------------
```


1.0
1.1
1.25
1.4
1.6
1.8
2.0
2.2
2.5
2.8
3.2
3.6
4.0
4.5
5.0
5.6
6.3

3 of 3

```
---------------------------------------------------------------------------------------------------------------

FUNCTION-CONTROLLED VARIABLE INPUT DATA FOR HEAT TRANSFER:

    COUNTER   ID #    ............NAME............   ARGUMENT F #    MULTIPLIER    L1     L2     L3

---------------------------------------------------------------------------------------------------------------

FUNCTION INPUT DATA FOR HEAT TRANSFER:

    COUNTER   F #    ID #     ............NAME............     MULTIPLIER       VALUE       L1     L2     L3     L4
---------------------------------------------------------------------------------------------------------------
---------------------------------------------------------------------------------------------------------------


---------------------------------------------------------------------------------------------------------------

FUNCTION-CONTROLLED VARIABLE INPUT DATA FOR REACTOR DYNAMICS:

    COUNTER   ID #    ............NAME............   ARGUMENT F #    MULTIPLIER    L1     L2     L3

---------------------------------------------------------------------------------------------------------------

FUNCTION INPUT DATA FOR REACTOR DYNAMICS:

    COUNTER   F #    ID #     ............NAME............     MULTIPLIER       VALUE       L1     L2     L3     L4
---------------------------------------------------------------------------------------------------------------

---------------------------------------------------------------------------------------------------------------

FUNCTION-CONTROLLED VARIABLE INPUT DATA FOR THE SYSTEM:

    COUNTER   ID #    ............NAME............   ARGUMENT F #    MULTIPLIER    L1     L2     L3

       1       35    MASS FLOW RATE BC AT NODE 1          10         1.0000E+00   101     1      0
---------------------------------------------------------------------------------------------------------------

FUNCTION INPUT DATA FOR THE SYSTEM:

    COUNTER   F #    ID #     ............NAME............     MULTIPLIER       VALUE       L1     L2     L3     L4

       11      5       1      PROBLEM TIME                     1.0000E+00    0.0000E+00    0      0      0      0
       12     10     114      TABLE, FLOWRATE BC VS TIME       1.0000E+00    1.0000E-03    0      0      0      0
                              TABLE ARGUMENT F #:      5
                              TABLE X,Y ENTRIES:
                                 0.0000E+00    5.0000E-04
                                 5.0000E+00    3.5900E-04
                                 2.0000E+01    3.5900E-04

---------------------------------------------------------------------------------------------------------------




  EXECUTION-TIME STATISTICS:

---------------------------------------------------------------------------------------------------------------

  * STEADY-STATE CALCULATION CONVERGED IN   74 WALL TEMPERATURE ITERATIONS,
    PROCEEDING WITH TRANSIENT CALCULATION...

 TIME (s) =   0.00000      TIME STEP #        0  TIME STEP (s) =   1.00000E-02
 TIME (s) =    30.000      TIME STEP #     3000  TIME STEP (s) =   1.00000E-02
 TIME (s) =    60.000      TIME STEP #     6000  TIME STEP (s) =   1.00000E-02
 TIME (s) =    90.000      TIME STEP #     9000  TIME STEP (s) =   1.00000E-02

  EXECUTION TIME:    234.21      s   (  3.9035      min)
```

PROGRAM SAFSIM OUTPUT (FLUID MECHANICS):

```
--------------------------------------------------------------------------------------------------------
EXAMPLE PROBLEM - INVESTIGATE STABILITY OF FLOW IN PARALLEL FLOW PATHS
CONTAINING POROUS BEDS WITH INTERNAL HEAT GENERATION
--------------------------------------------------------------------------------------------------------

v.v.v.v.v.  /.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v  /.v.v.v.v.v.v.v.v.v.v.v.v.v.
      ---> SYSTEM TIME (s) =   0.00000      TIME STEP #       0   TIME STEP (s) =   1.0000E-02 <---

><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><

 *** NETWORK #   1      NETWORK NAME: PARALLEL BEDS                        QUASI-STATIC SOLUTION

 * TIME (s) =   0.00000       SYSTEM TIME STEP #        0   SYSTEM TIME STEP (s) =   1.0000E-02
 * FLUID MECHANICS TIME STEP #        0   FLUID MECHANICS TIME STEP (s) =   1.0000E-02

 E# NODE1 NODE2    NAME         P1 (Pa)     P2 (Pa)  P2 - P1 (Pa)    T1 (K)      T2 (K)   T2 - T1 (K)  D1 (kg/cub.m) D2 (kg/cub.m)
  1    1     2  INLET        7.5791E+06  7.5791E+06 -2.9458E-05    125.000     125.000       0.000      1.4702E+01    1.4702E+01
  2    2     3  INLET        7.5791E+06  7.5791E+06 -2.9458E-05    125.000     125.000       0.000      1.4702E+01    1.4702E+01
  3    3     4  INLET        7.5791E+06  7.5791E+06 -2.9476E-05    125.000     125.000       0.000      1.4702E+01    1.4702E+01
  4    4     5  INLET        7.5791E+06  7.5791E+06 -3.2989E-05    125.000     125.001       0.001      1.4702E+01    1.4702E+01
  5    5     6  INLET        7.5791E+06  7.5791E+06 -8.5783E-02    125.001     125.187       0.186      1.4702E+01    1.4680E+01
 :
  6    6     7  BED1         7.5791E+06  7.5791E+06 -4.5093E+01    125.187     197.685      72.498      1.4680E+01    9.2962E+00
  7    7     8  BED1         7.5791E+06  7.5790E+06 -6.1872E+01    197.685     249.471      51.786      9.2962E+00    7.3664E+00
  8    8     9  BED1         7.5790E+06  7.5789E+06 -7.9503E+01    249.471     307.107      57.635      7.3664E+00    5.9839E+00
  9    9    10  BED1         7.5789E+06  7.5788E+06 -9.8240E+01    307.107     363.223      56.116      5.9839E+00    5.0593E+00
 10   10    11  BED1         7.5788E+06  7.5787E+06 -1.1790E+02    363.223     419.820      56.597      5.0593E+00    4.3772E+00
 :
 11   11    12  BED1         7.5787E+06  7.5786E+06 -1.3848E+02    419.820     476.334      56.514      4.3772E+00    3.8578E+00
 12   12    13  BED1         7.5786E+06  7.5784E+06 -1.5992E+02    476.334     532.909      56.575      3.8578E+00    3.4482E+00
 13   13    14  BED1         7.5784E+06  7.5782E+06 -1.8219E+02    532.909     589.487      56.579      3.4482E+00    3.1171E+00
 14   14    15  BED1         7.5782E+06  7.5780E+06 -2.0536E+02    589.487     646.386      56.898      3.1171E+00    2.8427E+00
 15   15    16  BED1         7.5780E+06  7.5778E+06 -2.2604E+02    646.386     692.241      45.855      2.8427E+00    2.6543E+00
 :
 16    6    17  BED2         7.5791E+06  7.5791E+06 -3.5189E+01    125.187     233.818     108.631      1.4680E+01    7.8596E+00
 17   17    18  BED2         7.5791E+06  7.5790E+06 -5.2633E+01    233.818     306.255      72.437      7.8596E+00    6.0006E+00
 18   18    19  BED2         7.5790E+06  7.5790E+06 -7.1522E+01    306.255     388.576      82.321      6.0006E+00    4.7293E+00
 19   19    20  BED2         7.5790E+06  7.5789E+06 -9.2195E+01    388.576     468.424      79.848      4.7293E+00    3.9231E+00
 20   20    21  BED2         7.5789E+06  7.5787E+06 -1.1437E+02    468.424     549.024      80.600      3.9231E+00    3.3471E+00
 :
 21   21    22  BED2         7.5787E+06  7.5786E+06 -1.3802E+02    549.024     629.487      80.464      3.3471E+00    2.9192E+00
 22   22    23  BED2         7.5786E+06  7.5784E+06 -1.6305E+02    629.487     710.028      80.541      2.9192E+00    2.5880E+00
 23   23    24  BED2         7.5784E+06  7.5783E+06 -1.8942E+02    710.028     790.560      80.532      2.5880E+00    2.3243E+00
 24   24    25  BED2         7.5783E+06  7.5780E+06 -2.1698E+02    790.560     870.712      80.152      2.3243E+00    2.1103E+00
 25   25    26  BED2         7.5780E+06  7.5778E+06 -2.4122E+02    870.712     933.235      62.523      2.1103E+00    1.9689E+00
 :
 E# NODE1 NODE2    NAME         P1 (Pa)     P2 (Pa)  P2 - P1 (Pa)    T1 (K)      T2 (K)   T2 - T1 (K)  D1 (kg/cub.m) D2 (kg/cub.m)
--------------------------------------------------------------------------------------------------------
 E#     NAME     FLOW (kg/s) VELOCITY (m/s)   P (Pa)        T (K)    effectv FF    Qflow (W)    Qconv (W) HTC (W/sq.m*K) Twall (K)
  1  INLET       5.0000E-04   1.7005E-01   7.5791E+06    125.000  1.9596E-02  1.0641E-14  0.0000E+00   0.0000E+00        0.000
  2  INLET       5.0000E-04   1.7005E-01   7.5791E+06    125.000  1.9596E-02  1.0641E-14  0.0000E+00   0.0000E+00        0.000
  3  INLET       5.0000E-04   1.7005E-01   7.5791E+06    125.000  1.9596E-02 -6.2281E-13  0.0000E+00   0.0000E+00        0.000
  4  INLET       5.0000E-04   1.7005E-01   7.5791E+06    125.000  1.9596E-02 -1.2009E-10  0.0000E+00   0.0000E+00        0.000
  5  INLET       5.0000E-04   1.7018E-01   7.5791E+06    125.094  5.6580E+01 -2.2656E-08  0.0000E+00   0.0000E+00        0.000
 :
  6  BED1        2.9399E-04   6.9385E-01   7.5791E+06    173.487  4.2266E+01 -5.5039E-05  3.1987E+02   1.0254E+04      208.149
  7  BED1        2.9399E-04   9.2715E-01   7.5790E+06    231.820  4.4319E+01 -5.4447E-05  2.2832E+02   9.7544E+03      257.828
  8  BED1        2.9399E-04   1.1485E+00   7.5790E+06    287.168  4.6116E+01 -7.5427E-05  2.5418E+02   9.4022E+03      317.206
  9  BED1        2.9399E-04   1.3741E+00   7.5789E+06    343.576  4.7836E+01 -8.8450E-05  2.4746E+02   9.1177E+03      373.733
 10  BED1        2.9399E-04   1.5991E+00   7.5788E+06    399.809  4.9461E+01 -1.0421E-04  2.4959E+02   8.8843E+03      431.024
 :
 11  BED1        2.9399E-04   1.8246E+00   7.5786E+06    456.185  5.1018E+01 -1.1909E-04  2.4922E+02   8.6863E+03      488.064
 12  BED1        2.9399E-04   2.0502E+00   7.5785E+06    512.593  5.2515E+01 -1.3427E-04  2.4949E+02   8.5152E+03      545.148
 13  BED1        2.9399E-04   2.2761E+00   7.5783E+06    569.041  5.3961E+01 -1.4934E-04  2.4950E+02   8.3649E+03      602.183
 14  BED1        2.9399E-04   2.5028E+00   7.5781E+06    625.708  5.5367E+01 -1.6538E-04  2.5100E+02   8.2308E+03      659.592
 15  BED1        2.9399E-04   2.7020E+00   7.5779E+06    675.501  5.6568E+01 -1.4473E-04  2.0258E+02   8.1242E+03      703.208
 :
 16  BED2        2.0601E-04   5.9178E-01   7.5791E+06    200.605  5.8087E+01 -3.4957E-05  3.3595E+02   8.6612E+03      243.703
 17  BED2        2.0601E-04   8.3617E-01   7.5791E+06    283.448  6.2981E+01 -3.5071E-05  2.2376E+02   8.1645E+03      313.900
 18  BED2        2.0601E-04   1.0683E+00   7.5790E+06    362.138  6.7214E+01 -5.1281E-05  2.5440E+02   7.8307E+03      398.235
 19  BED2        2.0601E-04   1.3050E+00   7.5789E+06    442.376  7.1234E+01 -6.1357E-05  2.4673E+02   7.5686E+03      478.597
 20  BED2        2.0601E-04   1.5411E+00   7.5788E+06    522.398  7.5013E+01 -7.3540E-05  2.4906E+02   7.3579E+03      560.008
 :
 21  BED2        2.0601E-04   1.7778E+00   7.5787E+06    602.625  7.8619E+01 -8.5051E-05  2.4864E+02   7.1817E+03      641.093
 22  BED2        2.0601E-04   2.0147E+00   7.5785E+06    682.896  8.2075E+01 -9.6780E-05  2.4888E+02   7.0310E+03      722.226
 23  BED2        2.0601E-04   2.2517E+00   7.5784E+06    763.217  8.5404E+01 -1.0843E-04  2.4885E+02   6.8998E+03      803.291
 24  BED2        2.0601E-04   2.4880E+00   7.5781E+06    843.308  8.8613E+01 -1.1950E-04  2.4781E+02   6.7842E+03      883.894
 25  BED2        2.0601E-04   2.6900E+00   7.5779E+06    911.744  9.1277E+01 -1.0079E-04  1.9368E+02   6.6952E+03      943.886
 :
 E#     NAME     FLOW (kg/s) VELOCITY (m/s)   P (Pa)        T (K)    effectv FF    Qflow (W)    Qconv (W) HTC (W/sq.m*K) Twall (K)
--------------------------------------------------------------------------------------------------------
 E#     NAME     DEN (kg/cub.m)    MACH #   DVISC (Pa*s) Cp (J/kg*K)    GAMMA  COND (W/m*K)   REYNOLDS #   PRANDTL #    PECLET #
  1  INLET         1.4702E+01    0.00020   5.2756E-06   1.5000E+04   1.37921  2.0000E-01   6.7006E+04  3.9567E-01  1.8750E+02
  2  INLET         1.4702E+01    0.00020   5.2756E-06   1.5000E+04   1.37921  2.0000E-01   6.7006E+04  3.9567E-01  1.8750E+02
  3  INLET         1.4702E+01    0.00020   5.2756E-06   1.5000E+04   1.37921  2.0000E-01   6.7006E+04  3.9567E-01  1.8750E+02
  4  INLET         1.4702E+01    0.00020   5.2756E-06   1.5000E+04   1.37921  2.0000E-01   6.7006E+04  3.9567E-01  1.8750E+02
  5  INLET         1.4691E+01    0.00020   5.2783E-06   1.5000E+04   1.37921  2.0000E-01   6.6972E+04  3.9587E-01  1.8750E+02
 :
  6  BED1          1.0593E+01    0.00070   6.5881E-06   1.5000E+04   1.37921  2.0000E-01   4.4625E+02  4.9411E-01  5.5124E+02
  7  BED1          7.9273E+00    0.00081   8.0183E-06   1.5000E+04   1.37921  2.0000E-01   3.6665E+02  6.0137E-01  5.5124E+02
  8  BED1          6.3994E+00   0.00090   9.2706E-06   1.5000E+04    1.37921  2.0000E-01    3.1713E+02  6.9529E-01  5.5124E+02
  9  BED1          5.3486E+00   0.00098   1.0469E-05   1.5000E+04    1.37921  2.0000E-01    2.8083E+02  7.8516E-01  5.5124E+02
 10  BED1          4.5963E+00   0.00106   1.1601E-05   1.5000E+04    1.37921  2.0000E-01    2.5341E+02  8.7011E-01  5.5124E+02
:
 11  BED1          4.0282E+00   0.00113   1.2687E-05   1.5000E+04    1.37921  2.0000E-01    2.3174E+02  9.5149E-01  5.5124E+02
 12  BED1          3.5849E+00   0.00120   1.3730E-05   1.5000E+04    1.37921  2.0000E-01    2.1413E+02  1.0297E+00  5.5124E+02
 13  BED1          3.2292E+00   0.00127   1.4737E-05   1.5000E+04    1.37921  2.0000E-01    1.9949E+02  1.1053E+00  5.5124E+02
 14  BED1          2.9367E+00   0.00133   1.5716E-05   1.5000E+04    1.37921  2.0000E-01    1.8706E+02  1.1787E+00  5.5124E+02
 15  BED1          2.7201E+00   0.00138   1.6554E-05   1.5000E+04    1.37921  2.0000E-01    1.7760E+02  1.2415E+00  5.5124E+02
:
 16  BED2          9.1609E+00   0.00055   7.2696E-06   1.5000E+04    1.37921  2.0000E-01    2.9830E+02  5.4522E-01  4.0659E+02
 17  BED2          6.4834E+00   0.00066   9.1890E-06   1.5000E+04    1.37921  2.0000E-01    2.3599E+02  6.8917E-01  4.0659E+02
 18  BED2          5.0746E+00   0.00074   1.0849E-05   1.5000E+04    1.37921  2.0000E-01    1.9988E+02  8.1366E-01  4.0659E+02
 19  BED2          4.1541E+00   0.00082   1.2425E-05   1.5000E+04    1.37921  2.0000E-01    1.7453E+02  9.3187E-01  4.0659E+02
 20  BED2          3.5177E+00   0.00089   1.3907E-05   1.5000E+04    1.37921  2.0000E-01    1.5593E+02  1.0430E+00  4.0659E+02
:
 21  BED2          3.0494E+00   0.00096   1.5321E-05   1.5000E+04    1.37921  2.0000E-01    1.4154E+02  1.1491E+00  4.0659E+02
 22  BED2          2.6909E+00   0.00102   1.6676E-05   1.5000E+04    1.37921  2.0000E-01    1.3003E+02  1.2507E+00  4.0659E+02
 23  BED2          2.4076E+00   0.00108   1.7982E-05   1.5000E+04    1.37921  2.0000E-01    1.2059E+02  1.3486E+00  4.0659E+02
 24  BED2          2.1789E+00   0.00114   1.9240E-05   1.5000E+04    1.37921  2.0000E-01    1.1271E+02  1.4430E+00  4.0659E+02
 25  BED2          2.0153E+00   0.00118   2.0285E-05   1.5000E+04    1.37921  2.0000E-01    1.0690E+02  1.5214E+00  4.0659E+02
:
 E#      NAME    DEN (kg/cub.m)    MACH #  DVISC (Pa*s) Cp (J/kg*K)    GAMMA  COND (W/m*K)  REYNOLDS #    PRANDTL #    PECLET #
---------------------------------------------------------------------------------------------------------------------------------
---------------------------------------------------------------------------------------------------------------------------------

v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.
      ---> SYSTEM TIME (s) =     30.000        TIME STEP #      3000   TIME STEP (s) =    1.0000E-02 <---

><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><

  *** NETWORK #    1      NETWORK NAME: PARALLEL BEDS                              QUASI-STATIC SOLUTION

  * TIME (s) =    30.000       SYSTEM TIME STEP #      3000   SYSTEM TIME STEP (s) =   1.0000E-02
  * FLUID MECHANICS TIME STEP #      3000   FLUID MECHANICS TIME STEP (s) =   1.0000E-02

  E# NODE1 NODE2     NAME         P1 (Pa)      P2 (Pa)  P2 - P1 (Pa)    T1 (K)       T2 (K)    T2 - T1 (K)  D1 (kg/cub.m) D2 (kg/cub.m)
   1     1     2  INLET         7.5790E+06  7.5790E+06 -1.6364E-05    125.000     125.000       0.000       1.4702E+01    1.4702E+01
   2     2     3  INLET         7.5790E+06  7.5790E+06 -1.6364E-05    125.000     125.000       0.000       1.4702E+01    1.4702E+01
   3     3     4  INLET         7.5790E+06  7.5790E+06 -1.6414E-05    125.000     125.000       0.000       1.4702E+01    1.4702E+01
   4     4     5  INLET         7.5790E+06  7.5790E+06 -2.3026E-05    125.000     125.004       0.004       1.4702E+01    1.4701E+01
   5     5     6  INLET         7.5790E+06  7.5790E+06 -4.4839E-02    125.004     125.494       0.491       1.4701E+01    1.4644E+01
:
   6     6     7  BED1          7.5790E+06  7.5790E+06 -3.9294E+01    125.494     206.709      81.214       1.4644E+01    8.8903E+00
   7     7     8  BED1          7.5790E+06  7.5789E+06 -5.5191E+01    206.709     263.897      57.188       8.8903E+00    6.9636E+00
   8     8     9  BED1          7.5789E+06  7.5789E+06 -7.2023E+01    263.897     327.809      63.912       6.9636E+00    5.6059E+00
   9     9    10  BED1          7.5789E+06  7.5788E+06 -9.0048E+01    327.809     389.996      62.187       5.6059E+00    4.7120E+00
  10    10    11  BED1          7.5788E+06  7.5787E+06 -1.0907E+02    389.996     452.725      62.728       4.7120E+00    4.0590E+00
:
  11    11    12  BED1          7.5787E+06  7.5785E+06 -1.2908E+02    452.725     515.360      62.636       4.0590E+00    3.5656E+00
  12    12    13  BED1          7.5785E+06  7.5784E+06 -1.5003E+02    515.360     578.062      62.702       3.5656E+00    3.1788E+00
  13    13    14  BED1          7.5784E+06  7.5782E+06 -1.7187E+02    578.062     640.772      62.709       3.1788E+00    2.8676E+00
  14    14    15  BED1          7.5782E+06  7.5780E+06 -1.9463E+02    640.772     703.697      62.925       2.8676E+00    2.6112E+00
  15    15    16  BED1          7.5780E+06  7.5778E+06 -2.1491E+02    703.697     754.071      50.373       2.6112E+00    2.4367E+00
:
  16     6    17  BED2          7.5790E+06  7.5790E+06 -1.8995E+01    125.494     390.340     264.845       1.4644E+01    4.7080E+00
  17    17    18  BED2          7.5790E+06  7.5790E+06 -3.5095E+01    390.340     544.138     153.798       4.7080E+00    3.3773E+00
  18    18    19  BED2          7.5790E+06  7.5789E+06 -5.3778E+01    544.138     724.667     180.529       3.3773E+00    2.5359E+00
  19    19    20  BED2          7.5789E+06  7.5788E+06 -7.5616E+01    724.667     899.463     174.796       2.5359E+00    2.0431E+00
  20    20    21  BED2          7.5788E+06  7.5787E+06 -1.0010E+02    899.463    1075.871     176.408       2.0431E+00    1.7080E+00
:
  21    21    22  BED2          7.5787E+06  7.5786E+06 -1.2717E+02   1075.871    1252.059     176.188       1.7080E+00    1.4677E+00
  22    22    23  BED2          7.5786E+06  7.5785E+06 -1.5664E+02   1252.059    1428.268     176.210       1.4677E+00    1.2866E+00
  23    23    24  BED2          7.5785E+06  7.5783E+06 -1.8831E+02   1428.268    1603.603     175.335       1.2866E+00    1.1459E+00
  24    24    25  BED2          7.5783E+06  7.5780E+06 -2.2125E+02   1603.603    1772.667     169.064       1.1459E+00    1.0366E+00
  25    25    26  BED2          7.5780E+06  7.5778E+06 -2.4920E+02   1772.667    1896.556     123.890       1.0366E+00    9.6881E-01
:
  E# NODE1 NODE2     NAME         P1 (Pa)      P2 (Pa)  P2 - P1 (Pa)    T1 (K)       T2 (K)    T2 - T1 (K)  D1 (kg/cub.m) D2 (kg/cub.m)
---------------------------------------------------------------------------------------------------------------------------------
  E#      NAME      FLOW (kg/s) VELOCITY (m/s)    P (Pa)          T (K)     effectv FF     Qflow (W)     Qconv (W) HTC (W/sq.m*K) Twall (K)
   1  INLET         3.5900E-04    1.2210E-01    7.5790E+06      125.000  2.1116E-02 -1.3599E-15   0.0000E+00    0.0000E+00        0.000
   2  INLET         3.5900E-04    1.2210E-01    7.5790E+06      125.000  2.1116E-02 -1.3595E-15   0.0000E+00    0.0000E+00        0.000
   3  INLET         3.5900E-04    1.2210E-01    7.5790E+06      125.000  2.1116E-02 -1.2066E-12   0.0000E+00    0.0000E+00        0.000
   4  INLET         3.5900E-04    1.2210E-01    7.5790E+06      125.002  2.1116E-02 -1.6267E-10   0.0000E+00    0.0000E+00        0.000
   5  INLET         3.5900E-04    1.2234E-01    7.5790E+06      125.249  5.6581E+01 -2.2106E-08   0.0000E+00    0.0000E+00        0.000
:
   6  BED1          2.6515E-04    6.4937E-01    7.5790E+06      180.027  4.3561E+01 -4.6537E-05   3.2320E+02    9.5888E+03      217.478
   7  BED1          2.6515E-04    8.8256E-01    7.5790E+06      244.673  4.6046E+01 -4.6423E-05   2.2740E+02    9.0963E+03      272.450
   8  BED1          2.6515E-04    1.1037E+00    7.5789E+06      305.980  4.8211E+01 -6.5232E-05   2.5421E+02    8.7544E+03      338.244
   9  BED1          2.6515E-04    1.3292E+00    7.5788E+06      368.484  5.0278E+01 -7.7001E-05   2.4733E+02    8.4806E+03      400.889
  10  BED1          2.6515E-04    1.5540E+00    7.5787E+06      430.798  5.2230E+01 -9.1190E-05   2.4949E+02    8.2574E+03      464.369
:
  11  BED1          2.6515E-04    1.7794E+00    7.5786E+06      493.273  5.4096E+01 -1.0461E-04   2.4912E+02    8.0690E+03      527.577
  12  BED1          2.6515E-04    2.0049E+00    7.5785E+06      555.783  5.5889E+01 -1.1828E-04   2.4938E+02    7.9067E+03      590.828
  13  BED1          2.6515E-04    2.2306E+00    7.5783E+06      618.342  5.7619E+01 -1.3187E-04   2.4941E+02    7.7645E+03      654.033
  14  BED1          2.6515E-04    2.4569E+00    7.5781E+06      681.057  5.9298E+01 -1.4597E-04   2.5037E+02    7.6380E+03      717.478
  15  BED1          2.6515E-04    2.6547E+00    7.5779E+06      735.862  6.0724E+01 -1.2688E-04   2.0075E+02    7.5383E+03      765.452
:
  16  BED2          9.3850E-05    4.3002E-01    7.5790E+06      319.971  9.5393E+01 -1.1580E-05   3.7368E+02    5.0544E+03      402.118
  17  BED2          9.3850E-05    6.7378E-01    7.5790E+06      501.349  1.1526E+02 -1.2185E-05   2.1631E+02    4.6851E+03      552.647
  18  BED2          9.3850E-05    9.0445E-01    7.5789E+06      672.980  1.3198E+02 -1.9423E-05   2.5418E+02    4.4582E+03      736.330
  19  BED2          9.3850E-05    1.1401E+00    7.5789E+06      848.318  1.4769E+02 -2.4078E-05   2.4606E+02    4.2876E+03      912.082
  20  BED2          9.3850E-05    1.3754E+00    7.5788E+06     1023.365  1.6235E+02 -2.9563E-05   2.4834E+02    4.1543E+03     1089.786
 :
 21  BED2        9.3850E-05   1.6113E+00   7.5787E+06    1198.877  1.7626E+02 -3.4807E-05  2.4803E+02   4.0451E+03     1267.005
 22  BED2        9.3850E-05   1.8473E+00   7.5785E+06    1374.445  1.8953E+02 -4.0095E-05  2.4807E+02   3.9532E+03     1444.168
 23  BED2        9.3850E-05   2.0826E+00   7.5784E+06    1549.497  2.0222E+02 -4.5146E-05  2.4688E+02   3.8743E+03     1620.298
 24  BED2        9.3850E-05   2.3119E+00   7.5782E+06    1720.038  2.1415E+02 -4.8501E-05  2.3834E+02   3.8069E+03     1789.603
 25  BED2        9.3850E-05   2.4970E+00   7.5779E+06    1857.744  2.2351E+02 -3.8276E-05  1.7535E+02   3.7579E+03     1909.589
 :
 E#     NAME     FLOW (kg/s) VELOCITY (m/s)   P (Pa)        T (K)     effectv FF   Qflow (W)    Qconv (W) HTC (W/sq.m*K) Twall (K)
------------------------------------------------------------------------------------------------------------------------------------
 E#     NAME     DEN (kg/cub.m)    MACH #  DVISC (Pa*s) Cp (J/kg*K)    GAMMA  COND (W/m*K)  REYNOLDS #    PRANDTL #    PECLET #
  1  INLET          1.4702E+01   0.00014   5.2756E-06   1.5000E+04    1.37921  2.0000E-01   4.8111E+04  3.9567E-01  1.3463E+02
  2  INLET          1.4702E+01   0.00014   5.2756E-06   1.5000E+04    1.37921  2.0000E-01   4.8111E+04  3.9567E-01  1.3463E+02
  3  INLET          1.4702E+01   0.00014   5.2756E-06   1.5000E+04    1.37921  2.0000E-01   4.8111E+04  3.9567E-01  1.3463E+02
  4  INLET          1.4701E+01   0.00014   5.2757E-06   1.5000E+04    1.37921  2.0000E-01   4.8110E+04  3.9568E-01  1.3463E+02
  5  INLET          1.4672E+01   0.00014   5.2827E-06   1.5000E+04    1.37921  2.0000E-01   4.8046E+04  3.9621E-01  1.3463E+02
 :
  6  BED1           1.0208E+01   0.00064   6.7554E-06   1.5000E+04    1.37921  2.0000E-01   3.9250E+02  5.0666E-01  4.9716E+02
  7  BED1           7.5108E+00   0.00075   8.3170E-06   1.5000E+04    1.37921  2.0000E-01   3.1880E+02  6.2377E-01  4.9716E+02
  8  BED1           6.0059E+00   0.00084   9.6780E-06   1.5000E+04    1.37921  2.0000E-01   2.7397E+02  7.2585E-01  4.9716E+02
  9  BED1           4.9871E+00   0.00092   1.0977E-05   1.5000E+04    1.37921  2.0000E-01   2.4154E+02  8.2330E-01  4.9716E+02
 10  BED1           4.2656E+00   0.00099   1.2204E-05   1.5000E+04    1.37921  2.0000E-01   2.1727E+02  9.1527E-01  4.9716E+02
 :
 11  BED1           3.7253E+00   0.00106   1.3377E-05   1.5000E+04    1.37921  2.0000E-01   1.9822E+02  1.0033E+00  4.9716E+02
 12  BED1           3.3063E+00   0.00113   1.4503E-05   1.5000E+04    1.37921  2.0000E-01   1.8282E+02  1.0878E+00  4.9716E+02
 13  BED1           2.9717E+00   0.00119   1.5591E-05   1.5000E+04    1.37921  2.0000E-01   1.7007E+02  1.1693E+00  4.9716E+02
 14  BED1           2.6980E+00   0.00125   1.6646E-05   1.5000E+04    1.37921  2.0000E-01   1.5929E+02  1.2484E+00  4.9716E+02
 15  BED1           2.4970E+00   0.00130   1.7542E-05   1.5000E+04    1.37921  2.0000E-01   1.5115E+02  1.3157E+00  4.9716E+02
 :
 16  BED2           5.7433E+00   0.00032   9.9757E-06   1.5000E+04    1.37921  2.0000E-01   9.9030E+01  7.4818E-01  1.8523E+02
 17  BED2           3.6655E+00   0.00040   1.3525E-05   1.5000E+04    1.37921  2.0000E-01   7.3043E+01  1.0144E+00  1.8523E+02
 18  BED2           2.7307E+00   0.00046   1.6512E-05   1.5000E+04    1.37921  2.0000E-01   5.9830E+01  1.2384E+00  1.8523E+02
 19  BED2           2.1663E+00   0.00052   1.9317E-05   1.5000E+04    1.37921  2.0000E-01   5.1140E+01  1.4488E+00  1.8523E+02
 20  BED2           1.7957E+00   0.00057   2.1937E-05   1.5000E+04    1.37921  2.0000E-01   4.5034E+01  1.6452E+00  1.8523E+02
 :
 21  BED2           1.5328E+00   0.00062   2.4421E-05   1.5000E+04    1.37921  2.0000E-01   4.0453E+01  1.8316E+00  1.8523E+02
 22  BED2           1.3370E+00   0.00066   2.6791E-05   1.5000E+04    1.37921  2.0000E-01   3.6874E+01  2.0093E+00  1.8523E+02
 23  BED2           1.1859E+00   0.00070   2.9059E-05   1.5000E+04    1.37921  2.0000E-01   3.3997E+01  2.1794E+00  1.8523E+02
 24  BED2           1.0683E+00   0.00074   3.1190E-05   1.5000E+04    1.37921  2.0000E-01   3.1674E+01  2.3392E+00  1.8523E+02
 25  BED2           9.8907E-01   0.00077   3.2861E-05   1.5000E+04    1.37921  2.0000E-01   3.0063E+01  2.4646E+00  1.8523E+02
 :
 E#     NAME     DEN (kg/cub.m)    MACH #  DVISC (Pa*s) Cp (J/kg*K)    GAMMA  COND (W/m*K)  REYNOLDS #    PRANDTL #    PECLET #
------------------------------------------------------------------------------------------------------------------------------------
------------------------------------------------------------------------------------------------------------------------------------

v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.
      ---> SYSTEM TIME (s) =     60.000        TIME STEP #      6000   TIME STEP (s) =   1.0000E-02 <---

><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><

  *** NETWORK #   1      NETWORK NAME: PARALLEL BEDS                                QUASI-STATIC SOLUTION

  * TIME (s) =    60.000        SYSTEM TIME STEP #     6000   SYSTEM TIME STEP (s) =   1.0000E-02
  * FLUID MECHANICS TIME STEP #      6000   FLUID MECHANICS TIME STEP (s) =   1.0000E-02

  E# NODE1 NODE2    NAME         P1 (Pa)     P2 (Pa)  P2 - P1 (Pa)   T1 (K)      T2 (K)    T2 - T1 (K)  D1 (kg/cub.m) D2 (kg/cub.m)
   1    1     2  INLET        7.5790E+06  7.5790E+06 -1.6364E-05    125.000     125.000      0.000        1.4702E+01    1.4702E+01
   2    2     3  INLET        7.5790E+06  7.5790E+06 -1.6364E-05    125.000     125.000      0.000        1.4702E+01    1.4702E+01
   3    3     4  INLET        7.5790E+06  7.5790E+06 -1.6416E-05    125.000     125.000      0.000        1.4702E+01    1.4702E+01
   4    4     5  INLET        7.5790E+06  7.5790E+06 -2.3479E-05    125.000     125.004      0.004        1.4702E+01    1.4701E+01
   5    5     6  INLET        7.5790E+06  7.5790E+06 -4.4907E-02    125.004     125.528      0.524        1.4701E+01    1.4640E+01
  :
   6    6     7  BED1         7.5790E+06  7.5790E+06 -4.0762E+01    125.528     204.322     78.794        1.4640E+01    8.9941E+00
   7    7     8  BED1         7.5790E+06  7.5790E+06 -5.6883E+01    204.322     260.021     55.700        8.9941E+00    7.0674E+00
   8    8     9  BED1         7.5790E+06  7.5789E+06 -7.3915E+01    260.021     322.199     62.178        7.0674E+00    5.7035E+00
   9    9    10  BED1         7.5789E+06  7.5788E+06 -9.2117E+01    322.199     382.709     60.510        5.7035E+00    4.8017E+00
  10   10    11  BED1         7.5788E+06  7.5787E+06 -1.1129E+02    382.709     443.743     61.034        4.8017E+00    4.1412E+00
  :
  11   11    12  BED1         7.5787E+06  7.5785E+06 -1.3144E+02    443.743     504.686     60.943        4.1412E+00    3.6410E+00
  12   12    13  BED1         7.5785E+06  7.5784E+06 -1.5250E+02    504.686     565.694     61.007        3.6410E+00    3.2483E+00
  13   13    14  BED1         7.5784E+06  7.5782E+06 -1.7445E+02    565.694     626.707     61.013        3.2483E+00    2.9320E+00
  14   14    15  BED1         7.5782E+06  7.5780E+06 -1.9730E+02    626.707     687.966     61.260        2.9320E+00    2.6709E+00
  15   15    16  BED1         7.5780E+06  7.5778E+06 -2.1767E+02    687.966     737.094     49.128        2.6709E+00    2.4928E+00
  :
  16    6    17  BED2         7.5790E+06  7.5790E+06 -1.8450E+01    125.528     416.724    291.196        1.4640E+01    4.4099E+00
  17   17    18  BED2         7.5790E+06  7.5790E+06 -3.4784E+01    416.724     583.039    166.315        4.4099E+00    3.1519E+00
  18   18    19  BED2         7.5790E+06  7.5789E+06 -5.3843E+01    583.039     778.874    195.835        3.1519E+00    2.3594E+00
  19   19    20  BED2         7.5789E+06  7.5789E+06 -7.6241E+01    778.874     968.488    189.613        2.3594E+00    1.8975E+00
  20   20    21  BED2         7.5789E+06  7.5788E+06 -1.0144E+02    968.488    1159.814    191.326        1.8975E+00    1.5844E+00
  :
  21   21    22  BED2         7.5788E+06  7.5786E+06 -1.2937E+02   1159.814    1350.895    191.082        1.5844E+00    1.3603E+00
  22   22    23  BED2         7.5786E+06  7.5785E+06 -1.5983E+02   1350.895    1541.938    191.043        1.3603E+00    1.1917E+00
  23   23    24  BED2         7.5785E+06  7.5783E+06 -1.9258E+02   1541.938    1731.803    189.865        1.1917E+00    1.0610E+00
  24   24    25  BED2         7.5783E+06  7.5781E+06 -2.2657E+02   1731.803    1914.056    182.253        1.0610E+00    9.5999E-01
  25   25    26  BED2         7.5781E+06  7.5778E+06 -2.5523E+02   1914.056    2046.634    132.577        9.5999E-01    8.9777E-01
  :
  E# NODE1 NODE2    NAME         P1 (Pa)     P2 (Pa)  P2 - P1 (Pa)   T1 (K)      T2 (K)    T2 - T1 (K)  D1 (kg/cub.m) D2 (kg/cub.m)
------------------------------------------------------------------------------------------------------------------------------------
  E#     NAME     FLOW (kg/s) VELOCITY (m/s)   P (Pa)        T (K)     effectv FF   Qflow (W)    Qconv (W) HTC (W/sq.m*K) Twall (K)
   1  INLET       3.5900E-04   1.2209E-01   7.5790E+06    125.000  2.1116E-02 -2.5350E-15  0.0000E+00   0.0000E+00        0.000
   2  INLET       3.5900E-04   1.2209E-01   7.5790E+06    125.000  2.1116E-02 -2.5346E-15  0.0000E+00   0.0000E+00        0.000
   3  INLET       3.5900E-04   1.2209E-01   7.5790E+06    125.000  2.1116E-02 -1.2760E-12  0.0000E+00   0.0000E+00        0.000
   4  INLET       3.5900E-04   1.2210E-01   7.5790E+06    125.002  2.1116E-02 -1.7372E-10  0.0000E+00   0.0000E+00        0.000
   5  INLET       3.5900E-04   1.2235E-01   7.5790E+06    125.266  5.6581E+01 -2.3611E-08  0.0000E+00   0.0000E+00        0.000
```

```
:
  6  BED1         2.7253E-04   6.6113E-01   7.5790E+06    178.324  4.3203E+01 -4.8679E-05  3.2229E+02   9.7606E+03     215.012
  7  BED1         2.7253E-04   8.9435E-01   7.5790E+06    241.228  4.5564E+01 -4.8442E-05  2.2765E+02   9.2665E+03     268.524
  8  BED1         2.7253E-04   1.1156E+00   7.5789E+06    300.890  4.7625E+01 -6.7805E-05  2.5419E+02   8.9220E+03     332.546
  9  BED1         2.7253E-04   1.3411E+00   7.5788E+06    361.710  4.9593E+01 -7.9893E-05  2.4736E+02   8.6454E+03     393.500
 10  BED1         2.7253E-04   1.5659E+00   7.5787E+06    422.343  5.1452E+01 -9.4482E-05  2.4950E+02   8.4196E+03     455.270
:
 11  BED1         2.7253E-04   1.7913E+00   7.5786E+06    483.132  5.3230E+01 -1.0827E-04  2.4913E+02   8.2286E+03     516.773
 12  BED1         2.7253E-04   2.0169E+00   7.5785E+06    543.955  5.4939E+01 -1.2233E-04  2.4939E+02   8.0641E+03     578.318
 13  BED1         2.7253E-04   2.2426E+00   7.5783E+06    604.823  5.6588E+01 -1.3629E-04  2.4942E+02   7.9198E+03     639.815
 14  BED1         2.7253E-04   2.4690E+00   7.5781E+06    665.866  5.8189E+01 -1.5089E-04  2.5052E+02   7.7913E+03     701.593
 15  BED1         2.7253E-04   2.6671E+00   7.5779E+06    719.289  5.9552E+01 -1.3137E-04  2.0123E+02   7.6898E+03     748.364
:
 16  BED2         8.6470E-05   4.2162E-01   7.5790E+06    340.500  1.0277E+02 -1.0469E-05  3.7864E+02   4.7684E+03     428.729
 17  BED2         8.6470E-05   6.6541E-01   7.5790E+06    537.383  1.2568E+02 -1.1028E-05  2.1549E+02   4.4150E+03     591.616
 18  BED2         8.6470E-05   8.9588E-01   7.5790E+06    723.507  1.4492E+02 -1.7692E-05  2.5406E+02   4.1992E+03     790.730
 19  BED2         8.6470E-05   1.1314E+00   7.5789E+06    913.671  1.6297E+02 -2.1985E-05  2.4593E+02   4.0374E+03     981.351
 20  BED2         8.6470E-05   1.3665E+00   7.5788E+06   1103.526  1.7982E+02 -2.7030E-05  2.4816E+02   3.9111E+03    1174.026
:
 21  BED2         8.6470E-05   1.6022E+00   7.5787E+06   1293.866  1.9580E+02 -3.1859E-05  2.4784E+02   3.8079E+03    1366.185
 22  BED2         8.6470E-05   1.8379E+00   7.5786E+06   1484.224  2.1103E+02 -3.6717E-05  2.4780E+02   3.7210E+03    1558.219
 23  BED2         8.6470E-05   2.0728E+00   7.5784E+06   1673.843  2.2559E+02 -4.1317E-05  2.4632E+02   3.6466E+03    1748.898
 24  BED2         8.6470E-05   2.3008E+00   7.5782E+06   1857.923  2.3922E+02 -4.4198E-05  2.3677E+02   3.5832E+03    1931.343
 25  BED2         8.6470E-05   2.4837E+00   7.5779E+06   2005.537  2.4984E+02 -3.4475E-05  1.7297E+02   3.5375E+03    2059.865
:
 E#      NAME    FLOW (kg/s) VELOCITY (m/s)   P (Pa)       T (K)    effectv FF   Qflow (W)    Qconv (W) HTC (W/sq.m*K) Twall (K)
----------------------------------------------------------------------------------------------------------------------------------
 E#      NAME    DEN (kg/cub.m)    MACH #  DVISC (Pa*s) Cp (J/kg*K)     GAMMA  COND (W/m*K)  REYNOLDS #   PRANDTL #   PECLET #
  1  INLET           1.4702E+01   0.00014   5.2756E-06   1.5000E+04    1.37921  2.0000E-01   4.8111E+04  3.9567E-01  1.3462E+02
  2  INLET           1.4702E+01   0.00014   5.2756E-06   1.5000E+04    1.37921  2.0000E-01   4.8111E+04  3.9567E-01  1.3462E+02
  3  INLET           1.4702E+01   0.00014   5.2756E-06   1.5000E+04    1.37921  2.0000E-01   4.8111E+04  3.9567E-01  1.3462E+02
  4  INLET           1.4701E+01   0.00014   5.2757E-06   1.5000E+04    1.37921  2.0000E-01   4.8110E+04  3.9568E-01  1.3462E+02
  5  INLET           1.4670E+01   0.00014   5.2832E-06   1.5000E+04    1.37921  2.0000E-01   4.8041E+04  3.9624E-01  1.3462E+02
:
  6  BED1            1.0305E+01   0.00066   6.7120E-06   1.5000E+04    1.37921  2.0000E-01   4.0603E+02  5.0340E-01  5.1099E+02
  7  BED1            7.6181E+00   0.00076   8.2374E-06   1.5000E+04    1.37921  2.0000E-01   3.3084E+02  6.1781E-01  5.1099E+02
  8  BED1            6.1075E+00   0.00085   9.5685E-06   1.5000E+04    1.37921  2.0000E-01   2.8482E+02  7.1764E-01  5.1099E+02
  9  BED1            5.0805E+00   0.00093   1.0840E-05   1.5000E+04    1.37921  2.0000E-01   2.5141E+02  8.1301E-01  5.1099E+02
 10  BED1            4.3510E+00   0.00101   1.2041E-05   1.5000E+04    1.37921  2.0000E-01   2.2634E+02  9.0306E-01  5.1099E+02
:
 11  BED1            3.8035E+00   0.00108   1.3190E-05   1.5000E+04    1.37921  2.0000E-01   2.0662E+02  9.8923E-01  5.1099E+02
 12  BED1            3.3782E+00   0.00115   1.4294E-05   1.5000E+04    1.37921  2.0000E-01   1.9067E+02  1.0720E+00  5.1099E+02
 13  BED1            3.0381E+00   0.00121   1.5359E-05   1.5000E+04    1.37921  2.0000E-01   1.7744E+02  1.1519E+00  5.1099E+02
 14  BED1            2.7595E+00   0.00127   1.6393E-05   1.5000E+04    1.37921  2.0000E-01   1.6625E+02  1.2295E+00  5.1099E+02
 15  BED1            2.5545E+00   0.00132   1.7274E-05   1.5000E+04    1.37921  2.0000E-01   1.5777E+02  1.2955E+00  5.1099E+02
:
 16  BED2            5.3971E+00   0.00030   1.0405E-05   1.5000E+04    1.37921  2.0000E-01   8.7477E+01  7.8039E-01  1.7066E+02
 17  BED2            3.4197E+00   0.00038   1.4176E-05   1.5000E+04    1.37921  2.0000E-01   6.4207E+01  1.0632E+00  1.7066E+02
 18  BED2            2.5400E+00   0.00044   1.7342E-05   1.5000E+04    1.37921  2.0000E-01   5.2485E+01  1.3007E+00  1.7066E+02
 19  BED2            2.0113E+00   0.00050   2.0314E-05   1.5000E+04    1.37921  2.0000E-01   4.4807E+01  1.5235E+00  1.7066E+02
 20  BED2            1.6653E+00   0.00055   2.3087E-05   1.5000E+04    1.37921  2.0000E-01   3.9425E+01  1.7315E+00  1.7066E+02
:
 21  BED2            1.4203E+00   0.00059   2.5716E-05   1.5000E+04    1.37921  2.0000E-01   3.5394E+01  1.9287E+00  1.7066E+02
 22  BED2            1.2381E+00   0.00063   2.8223E-05   1.5000E+04    1.37921  2.0000E-01   3.2250E+01  2.1168E+00  1.7066E+02
 23  BED2            1.0978E+00   0.00067   3.0620E-05   1.5000E+04    1.37921  2.0000E-01   2.9726E+01  2.2965E+00  1.7066E+02
 24  BED2            9.8901E-01   0.00071   3.2863E-05   1.5000E+04    1.37921  2.0000E-01   2.7697E+01  2.4647E+00  1.7066E+02
 25  BED2            9.1618E-01   0.00074   3.4611E-05   1.5000E+04    1.37921  2.0000E-01   2.6298E+01  2.5958E+00  1.7066E+02
:
 E#      NAME    DEN (kg/cub.m)    MACH #  DVISC (Pa*s) Cp (J/kg*K)     GAMMA  COND (W/m*K)  REYNOLDS #   PRANDTL #   PECLET #
----------------------------------------------------------------------------------------------------------------------------------
----------------------------------------------------------------------------------------------------------------------------------

v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.
        ---> SYSTEM TIME (s) =    90.000        TIME STEP #      9000   TIME STEP (s) =   1.0000E-02 <---

><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><

   *** NETWORK #   1      NETWORK NAME: PARALLEL BEDS                           QUASI-STATIC SOLUTION

   * TIME (s) =    90.000        SYSTEM TIME STEP #     9000   SYSTEM TIME STEP (s) =   1.0000E-02
   * FLUID MECHANICS TIME STEP #      9000   FLUID MECHANICS TIME STEP (s) =   1.0000E-02

   E# NODE1 NODE2    NAME          P1 (Pa)     P2 (Pa)  P2 - P1 (Pa)    T1 (K)      T2 (K)    T2 - T1 (K)  D1 (kg/cub.m) D2 (kg/cub.m)
    1     1     2  INLET        7.5792E+06  7.5792E+06 -1.6363E-05    125.000     125.000        0.000       1.4702E+01    1.4702E+01
    2     2     3  INLET        7.5792E+06  7.5792E+06 -1.6365E-05    125.000     125.000        0.000       1.4702E+01    1.4702E+01
    3     3     4  INLET        7.5792E+06  7.5792E+06 -1.6503E-05    125.000     125.000        0.000       1.4702E+01    1.4702E+01
    4     4     5  INLET        7.5792E+06  7.5792E+06 -3.5292E-05    125.000     125.010        0.010       1.4702E+01    1.4701E+01
    5     5     6  INLET        7.5792E+06  7.5792E+06 -4.6668E-02    125.010     126.408        1.397       1.4701E+01    1.4538E+01
  :
    6     6     7  BED1         7.5792E+06  7.5792E+06 -5.3384E+01    126.408     190.434       64.026       1.4538E+01    9.6503E+00
    7     7     8  BED1         7.5792E+06  7.5791E+06 -7.1346E+01    190.434     236.923       46.489       9.6503E+00    7.7566E+00
    8     8     9  BED1         7.5791E+06  7.5790E+06 -9.0083E+01    236.923     288.449       51.526       7.7566E+00    6.3710E+00
    9     9    10  BED1         7.5790E+06  7.5789E+06 -1.0985E+02    288.449     338.686       50.237       6.3710E+00    5.4259E+00
   10    10    11  BED1         7.5789E+06  7.5788E+06 -1.3047E+02    338.686     389.372       50.686       5.4259E+00    4.7195E+00
  :
   11    11    12  BED1         7.5788E+06  7.5786E+06 -1.5196E+02    389.372     440.013       50.641       4.7195E+00    4.1763E+00
   12    12    13  BED1         7.5786E+06  7.5785E+06 -1.7425E+02    440.013     490.739       50.725       4.1763E+00    3.7445E+00
   13    13    14  BED1         7.5785E+06  7.5783E+06 -1.9733E+02    490.739     541.489       50.750       3.7445E+00    3.3935E+00
   14    14    15  BED1         7.5783E+06  7.5780E+06 -2.2130E+02    541.489     592.665       51.176       3.3935E+00    3.1003E+00
   15    15    16  BED1         7.5780E+06  7.5778E+06 -2.4274E+02    592.665     634.224       41.559       3.1003E+00    2.8971E+00
  :
   16     6    17  BED2         7.5792E+06  7.5792E+06 -1.9890E+01    126.408    1056.484      930.076       1.4538E+01    1.7395E+00
   17    17    18  BED2         7.5792E+06  7.5792E+06 -4.1225E+01   1056.484    1440.302      383.819       1.7395E+00    1.2759E+00
 18   18    19  BED2        7.5792E+06  7.5791E+06 -6.4946E+01   1440.302   1896.366    456.064       1.2759E+00    9.6908E-01
 19   19    20  BED2        7.5791E+06  7.5790E+06 -9.2392E+01   1896.366   2323.238    426.872       9.6908E-01    7.9101E-01
 20   20    21  BED2        7.5790E+06  7.5789E+06 -1.2243E+02   2323.238   2738.540    415.302       7.9101E-01    6.7104E-01
:
 21   21    22  BED2        7.5789E+06  7.5787E+06 -1.5466E+02   2738.540   3139.660    401.121       6.7104E-01    5.8530E-01
 22   22    23  BED2        7.5787E+06  7.5786E+06 -1.8850E+02   3139.660   3525.343    385.682       5.8530E-01    5.2125E-01
 23   23    24  BED2        7.5786E+06  7.5783E+06 -2.2297E+02   3525.343   3888.280    362.937       5.2125E-01    4.7258E-01
 24   24    25  BED2        7.5783E+06  7.5781E+06 -2.5567E+02   3888.280   4205.597    317.317       4.7258E-01    4.3691E-01
 25   25    26  BED2        7.5781E+06  7.5778E+06 -2.8003E+02   4205.597   4411.150    205.553       4.3691E-01    4.1654E-01
:
 E# NODE1 NODE2    NAME        P1 (Pa)      P2 (Pa)  P2 - P1 (Pa)   T1 (K)      T2 (K)    T2 - T1 (K)  D1 (kg/cub.m) D2 (kg/cub.m)
----------------------------------------------------------------------------------------------------------------------------------
 E#      NAME      FLOW (kg/s) VELOCITY (m/s)   P (Pa)         T (K)    effectv FF    Qflow (W)     Qconv (W) HTC (W/sq.m*K) Twall (K)
  1   INLET        3.5900E-04   1.2209E-01   7.5792E+06     125.000  2.1116E-02  9.9809E-15  0.0000E+00   0.0000E+00         0.000
  2   INLET        3.5900E-04   1.2209E-01   7.5792E+06     125.000  2.1116E-02 -3.5501E-14  0.0000E+00   0.0000E+00         0.000
  3   INLET        3.5900E-04   1.2209E-01   7.5792E+06     125.000  2.1116E-02 -3.4011E-12  0.0000E+00   0.0000E+00         0.000
  4   INLET        3.5900E-04   1.2210E-01   7.5792E+06     125.005  2.1116E-02 -4.6221E-10  0.0000E+00   0.0000E+00         0.000
  5   INLET        3.5900E-04   1.2278E-01   7.5792E+06     125.709  5.6581E+01 -6.3169E-08  0.0000E+00   0.0000E+00         0.000
:
  6   BED1         3.2944E-04   7.5595E-01   7.5792E+06     168.682  4.1090E+01 -6.7112E-05  3.1653E+02   1.1017E+04       200.604
  7   BED1         3.2944E-04   9.8965E-01   7.5792E+06     220.829  4.2748E+01 -6.5727E-05  2.2969E+02   1.0516E+04       245.097
  8   BED1         3.2944E-04   1.2117E+00   7.5791E+06     270.368  4.4209E+01 -8.9559E-05  2.5463E+02   1.0156E+04       298.225
  9   BED1         3.2944E-04   1.4380E+00   7.5790E+06     320.861  4.5612E+01 -1.0424E-04  2.4825E+02   9.8618E+03       348.831
 10   BED1         3.2944E-04   1.6637E+00   7.5789E+06     371.223  4.6941E+01 -1.2210E-04  2.5047E+02   9.6183E+03       400.157
:
 11   BED1         3.2944E-04   1.8901E+00   7.5787E+06     421.738  4.8217E+01 -1.3898E-04  2.5024E+02   9.4104E+03       451.285
 12   BED1         3.2944E-04   2.1168E+00   7.5785E+06     472.309  4.9446E+01 -1.5624E-04  2.5066E+02   9.2299E+03       502.484
 13   BED1         3.2944E-04   2.3438E+00   7.5784E+06     522.940  5.0635E+01 -1.7330E-04  2.5078E+02   9.0707E+03       553.659
 14   BED1         3.2944E-04   2.5721E+00   7.5782E+06     573.860  5.1793E+01 -1.9211E-04  2.5296E+02   8.9280E+03       605.342
 15   BED1         3.2944E-04   2.7740E+00   7.5779E+06     618.887  5.2790E+01 -1.6886E-04  2.0570E+02   8.8137E+03       644.818
:
 16   BED2         2.9565E-05   3.6346E-01   7.5792E+06     858.526  3.8560E+02 -2.9165E-06  4.1661E+02   2.1866E+03      1070.227
 17   BED2         2.9565E-05   5.7312E-01   7.5792E+06    1353.759  5.1077E+02 -2.5450E-06  1.6966E+02   2.0258E+03      1446.817
 18   BED2         2.9565E-05   7.5780E-01   7.5791E+06    1789.972  6.0899E+02 -4.0443E-06  2.0247E+02   1.9331E+03      1906.350
 19   BED2         2.9565E-05   9.4032E-01   7.5791E+06    2221.073  6.9868E+02 -4.7946E-06  1.8939E+02   1.8645E+03      2333.941
 20   BED2         2.9565E-05   1.1165E+00   7.5790E+06    2637.144  7.8003E+02 -5.6049E-06  1.8428E+02   1.8116E+03      2750.171
:
 21   BED2         2.9565E-05   1.2871E+00   7.5788E+06    3040.127  8.5495E+02 -6.2988E-06  1.7800E+02   1.7689E+03      3151.935
 22   BED2         2.9565E-05   1.4515E+00   7.5787E+06    3428.341  9.2415E+02 -6.8820E-06  1.7121E+02   1.7337E+03      3538.070
 23   BED2         2.9565E-05   1.6072E+00   7.5784E+06    3795.963  9.8738E+02 -7.2228E-06  1.6130E+02   1.7043E+03      3901.119
 24   BED2         2.9565E-05   1.7462E+00   7.5782E+06    4124.148  1.0422E+03 -6.9221E-06  1.4157E+02   1.6808E+03      4217.733
 25   BED2         2.9565E-05   1.8453E+00   7.5779E+06    4358.072  1.0804E+03 -4.3805E-06  9.2719E+01   1.6654E+03      4419.933
:
 E#      NAME      FLOW (kg/s) VELOCITY (m/s)   P (Pa)         T (K)    effectv FF    Qflow (W)     Qconv (W) HTC (W/sq.m*K) Twall (K)
----------------------------------------------------------------------------------------------------------------------------------
 E#      NAME      DEN (kg/cub.m)   MACH #   DVISC (Pa*s) Cp (J/kg*K)     GAMMA  COND (W/m*K)   REYNOLDS #    PRANDTL #    PECLET #
  1   INLET          1.4702E+01   0.00014    5.2756E-06   1.5000E+04    1.37921  2.0000E-01    4.8111E+04  3.9567E-01  1.3463E+02
  2   INLET          1.4702E+01   0.00014    5.2756E-06   1.5000E+04    1.37921  2.0000E-01    4.8111E+04  3.9567E-01  1.3463E+02
  3   INLET          1.4702E+01   0.00014    5.2756E-06   1.5000E+04    1.37921  2.0000E-01    4.8111E+04  3.9567E-01  1.3463E+02
  4   INLET          1.4701E+01   0.00014    5.2758E-06   1.5000E+04    1.37921  2.0000E-01    4.8109E+04  3.9568E-01  1.3463E+02
  5   INLET          1.4619E+01   0.00015    5.2959E-06   1.5000E+04    1.37921  2.0000E-01    4.7926E+04  3.9719E-01  1.3463E+02
:
  6   BED1           1.0895E+01   0.00077    6.4639E-06   1.5000E+04    1.37921  2.0000E-01    5.0966E+02  4.8479E-01  6.1769E+02
  7   BED1           8.3220E+00   0.00088    7.7586E-06   1.5000E+04    1.37921  2.0000E-01    4.2461E+02  5.8190E-01  6.1769E+02
  8   BED1           6.7971E+00   0.00098    8.8994E-06   1.5000E+04    1.37921  2.0000E-01    3.7018E+02  6.6745E-01  6.1769E+02
  9   BED1           5.7274E+00   0.00106    9.9945E-06   1.5000E+04    1.37921  2.0000E-01    3.2962E+02  7.4959E-01  6.1769E+02
 10   BED1           4.9503E+00   0.00114    1.1033E-05   1.5000E+04    1.37921  2.0000E-01    2.9860E+02  8.2744E-01  6.1769E+02
:
 11   BED1           4.3573E+00   0.00122    1.2029E-05   1.5000E+04    1.37921  2.0000E-01    2.7387E+02  9.0218E-01  6.1769E+02
 12   BED1           3.8907E+00   0.00129    1.2989E-05   1.5000E+04    1.37921  2.0000E-01    2.5363E+02  9.7416E-01  6.1769E+02
 13   BED1           3.5139E+00   0.00136    1.3917E-05   1.5000E+04    1.37921  2.0000E-01    2.3672E+02  1.0438E+00  6.1769E+02
 14   BED1           3.2020E+00   0.00142    1.4822E-05   1.5000E+04    1.37921  2.0000E-01    2.2227E+02  1.1116E+00  6.1769E+02
 15   BED1           2.9689E+00   0.00148    1.5600E-05   1.5000E+04    1.37921  2.0000E-01    2.1117E+02  1.1700E+00  6.1769E+02
:
 16   BED2           2.1406E+00   0.00016    1.9475E-05   1.5000E+04    1.37921  2.0000E-01    1.5980E+01  1.4606E+00  5.8352E+01
 17   BED2           1.3575E+00   0.00021    2.6517E-05   1.5000E+04    1.37921  2.0000E-01    1.1736E+01  1.9888E+00  5.8352E+01
 18   BED2           1.0267E+00   0.00024    3.2044E-05   1.5000E+04    1.37921  2.0000E-01    9.7120E+00  2.4033E+00  5.8352E+01
 19   BED2           8.2740E-01   0.00026    3.7091E-05   1.5000E+04    1.37921  2.0000E-01    8.3905E+00  2.7818E+00  5.8352E+01
 20   BED2           6.9685E-01   0.00029    4.1668E-05   1.5000E+04    1.37921  2.0000E-01    7.4687E+00  3.1251E+00  5.8352E+01
:
 21   BED2           6.0447E-01   0.00031    4.5884E-05   1.5000E+04    1.37921  2.0000E-01    6.7824E+00  3.4413E+00  5.8352E+01
 22   BED2           5.3601E-01   0.00033    4.9778E-05   1.5000E+04    1.37921  2.0000E-01    6.2519E+00  3.7334E+00  5.8352E+01
 23   BED2           4.8408E-01   0.00035    5.3336E-05   1.5000E+04    1.37921  2.0000E-01    5.8348E+00  4.0002E+00  5.8352E+01
 24   BED2           4.4555E-01   0.00036    5.6420E-05   1.5000E+04    1.37921  2.0000E-01    5.5159E+00  4.2315E+00  5.8352E+01
 25   BED2           4.2162E-01   0.00037    5.8569E-05   1.5000E+04    1.37921  2.0000E-01    5.3135E+00  4.3927E+00  5.8352E+01
:
 E#      NAME      DEN (kg/cub.m)   MACH #   DVISC (Pa*s) Cp (J/kg*K)     GAMMA  COND (W/m*K)   REYNOLDS #    PRANDTL #    PECLET #
----------------------------------------------------------------------------------------------------------------------------------
----------------------------------------------------------------------------------------------------------------------------------

 * PROGRAM EXECUTED SUCCESSFULLY *

                                   SAFSIM: COPYRIGHT 1993, SANDIA CORPORATION
 PROGRAM SAFSIM OUTPUT (HEAT TRANSFER):

-------------------------------------------------------------------------------------------------------------------
 EXAMPLE PROBLEM - INVESTIGATE STABILITY OF FLOW IN PARALLEL FLOW PATHS
 CONTAINING POROUS BEDS WITH INTERNAL HEAT GENERATION
-------------------------------------------------------------------------------------------------------------------

v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.
      ---> SYSTEM TIME (s) =   0.00000       TIME STEP #        0   TIME STEP (s) =   1.0000E-02 <---

><><><><:<><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><

   *** STRUCTURE #    1   STRUCTURE NAME: BED1                      NUMBER OF COPIES =   1.0000E+00      DYNAMIC SOLUTION

   * TIME (s) =   0.00000       SYSTEM TIME STEP #        0   SYSTEM TIME STEP (s) =   1.0000E-02
   * HEAT TRANSFER TIME STEP #        0   HEAT TRANSFER TIME STEP (s) =   1.0000E+00   EXPLICIT LIMIT (s) =   1.9549E-02

   HEAT FLUX BOUNDARY CONDITION AT END 1:         HEAT FLUX (W/sq.m) =   0.0000E+00   Q (W) =   0.0000E+00
-------------------------------------------------------------------------------------------------------------------
  e/n#   I FAC    Tn (K)  dT/dt (K/s)    Te (K)  dT/dX (K/m)    POWER (W)    MATERIAL   COND (W/m*K)  C (J/kg*K)  DEN (kg/cub.m)
     1  1.0000   717.893 -5.3755E-02   703.178 -2.9431E+04  2.5000E+02  BED,D=400C    2.0000E+00   3.6000E+02    3.5580E+03
     2  1.0000   688.462 -5.0462E-02   659.565 -5.7795E+04  2.5000E+02  BED,D=400C    2.0000E+00   3.6000E+02    3.5580E+03
     3  1.0000   630.667 -4.3970E-02   602.159 -5.7016E+04  2.5000E+02  BED,D=400C    2.0000E+00   3.6000E+02    3.5580E+03
     4  1.0000   573.651 -3.7724E-02   545.128 -5.7047E+04  2.5000E+02  BED,D=400C    2.0000E+00   3.6000E+02    3.5580E+03
     5  1.0000   516.604 -3.1833E-02   488.047 -5.7113E+04  2.5000E+02  BED,D=400C    2.0000E+00   3.6000E+02    3.5580E+03
  :
     6  1.0000   459.491 -2.6276E-02   431.010 -5.6962E+04  2.5000E+02  BED,D=400C    2.0000E+00   3.6000E+02    3.5580E+03
     7  1.0000   402.529 -2.1051E-02   373.722 -5.7614E+04  2.5000E+02  BED,D=400C    2.0000E+00   3.6000E+02    3.5580E+03
     8  1.0000   344.915 -1.6102E-02   317.198 -5.5435E+04  2.5000E+02  BED,D=400C    2.0000E+00   3.6000E+02    3.5580E+03
     9  1.0000   289.481 -1.1589E-02   257.823 -6.3316E+04  2.5000E+02  BED,D=400C    2.0000E+00   3.6000E+02    3.5580E+03
    10  1.0000   226.165 -6.8546E-03   208.146 -3.6038E+04  2.5000E+02  BED,D=400C    2.0000E+00   3.6000E+02    3.5580E+03
  :
    11  1.0000   190.126 -4.3275E-03
-------------------------------------------------------------------------------------------------------------------
   e#    SURFACE    Q'' (W/sq.m)       Q (W)
-------------------------------------------------------------------------------------------------------------------
   e#    SURFACE  NW #     E#    REYNOLDS #  HTC (W/sq.m*K)  Twall (K)  Tfluid (K)  FILMDROP (K)  Q'' (W/sq.m)     Q (W)
    1       1       1      15   1.7759E+02    8.1242E+03     703.178     675.501       27.676       2.2485E+05   2.0236E+02
    2       1       1      14   1.8705E+02    8.2308E+03     659.565     625.708       33.857       2.7867E+05   2.5080E+02
    3       1       1      13   1.9948E+02    8.3649E+03     602.159     569.041       33.118       2.7703E+05   2.4933E+02
    4       1       1      12   2.1412E+02    8.5152E+03     545.128     512.593       32.535       2.7704E+05   2.4934E+02
    5       1       1      11   2.3173E+02    8.6863E+03     488.047     456.185       31.862       2.7676E+05   2.4909E+02
  :
    6       1       1      10   2.5340E+02    8.8843E+03     431.010     399.809       31.201       2.7720E+05   2.4948E+02
    7       1       1       9   2.8082E+02    9.1177E+03     373.722     343.576       30.146       2.7486E+05   2.4738E+02
    8       1       1       8   3.1712E+02    9.4022E+03     317.198     287.168       30.030       2.8235E+05   2.5411E+02
    9       1       1       7   3.6665E+02    9.7544E+03     257.823     231.820       26.003       2.5364E+05   2.2828E+02
   10       1       1       6   4.4624E+02    1.0254E+04     208.146     173.487       34.659       3.5538E+05   3.1984E+02
  :
-------------------------------------------------------------------------------------------------------------------
   HEAT FLUX BOUNDARY CONDITION AT END 2:         HEAT FLUX (W/sq.m) =   0.0000E+00   Q (W) =   0.0000E+00
-------------------------------------------------------------------------------------------------------------------
   STRUCTURE TOTALS: INTERNAL POWER (W) =   2.5000E+03   EXCHANGE POWER (W) =   2.5000E+03   NET POWER (W) =   4.5475E-13

><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><

   *** STRUCTURE #    2   STRUCTURE NAME: BED2                      NUMBER OF COPIES =   1.0000E+00      DYNAMIC SOLUTION

   * TIME (s) =   0.00000       SYSTEM TIME STEP #        0   SYSTEM TIME STEP (s) =   1.0000E-02
   * HEAT TRANSFER TIME STEP #        0   HEAT TRANSFER TIME STEP (s) =   1.0000E+00   EXPLICIT LIMIT (s) =   2.2886E-02

   HEAT FLUX BOUNDARY CONDITION AT END 1:         HEAT FLUX (W/sq.m) =   0.0000E+00   Q (W) =   0.0000E+00
-------------------------------------------------------------------------------------------------------------------
  e/n#   I FAC    Tn (K)  dT/dt (K/s)    Te (K)  dT/dX (K/m)    POWER (W)    MATERIAL   COND (W/m*K)  C (J/kg*K)  DEN (kg/cub.m)
     1  1.0000   963.863  1.1840E-01   943.953 -3.9821E+04  2.5000E+02  BED,D=400C    2.0000E+00   3.6000E+02    3.5580E+03
     2  1.0000   924.043  1.1125E-01   883.955 -8.0176E+04  2.5000E+02  BED,D=400C    2.0000E+00   3.6000E+02    3.5580E+03
     3  1.0000   843.867  9.6845E-02   803.343 -8.1048E+04  2.5000E+02  BED,D=400C    2.0000E+00   3.6000E+02    3.5580E+03
     4  1.0000   762.819  8.2695E-02   722.270 -8.1096E+04  2.5000E+02  BED,D=400C    2.0000E+00   3.6000E+02    3.5580E+03
     5  1.0000   681.722  6.9433E-02   641.130 -8.1185E+04  2.5000E+02  BED,D=400C    2.0000E+00   3.6000E+02    3.5580E+03
  :
     6  1.0000   600.537  5.7048E-02   560.038 -8.0997E+04  2.5000E+02  BED,D=400C    2.0000E+00   3.6000E+02    3.5580E+03
     7  1.0000   519.539  4.5517E-02   478.621 -8.1838E+04  2.5000E+02  BED,D=400C    2.0000E+00   3.6000E+02    3.5580E+03
     8  1.0000   437.702  3.4731E-02   398.252 -7.8899E+04  2.5000E+02  BED,D=400C    2.0000E+00   3.6000E+02    3.5580E+03
     9  1.0000   358.803  2.4963E-02   313.911 -8.9783E+04  2.5000E+02  BED,D=400C    2.0000E+00   3.6000E+02    3.5580E+03
    10  1.0000   269.020  1.4925E-02   243.711 -5.0619E+04  2.5000E+02  BED,D=400C    2.0000E+00   3.6000E+02    3.5580E+03
  :
    11  1.0000   218.401  9.6669E-03
-------------------------------------------------------------------------------------------------------------------
   e#    SURFACE    Q'' (W/sq.m)       Q (W)
-------------------------------------------------------------------------------------------------------------------
   e#    SURFACE  NW #     E#    REYNOLDS #  HTC (W/sq.m*K)  Twall (K)  Tfluid (K)  FILMDROP (K)  Q'' (W/sq.m)     Q (W)
    1       1       1      25   1.0691E+02    6.6952E+03     943.953     911.744       32.209       2.1564E+05   1.9408E+02
    2       1       1      24   1.1272E+02    6.7842E+03     883.955     843.308       40.647       2.7576E+05   2.4818E+02
    3       1       1      23   1.2060E+02    6.8998E+03     803.343     763.217       40.126       2.7686E+05   2.4918E+02
    4       1       1      22   1.3004E+02    7.0310E+03     722.270     682.896       39.374       2.7684E+05   2.4916E+02
    5       1       1      21   1.4155E+02    7.1817E+03     641.130     602.625       38.505       2.7653E+05   2.4888E+02
  :
    6       1       1      20   1.5594E+02    7.3579E+03     560.038     522.398       37.641       2.7696E+05   2.4926E+02
    7       1       1      19   1.7454E+02    7.5686E+03     478.621     442.376       36.245       2.7432E+05   2.4689E+02
    8       1       1      18   1.9989E+02    7.8307E+03     398.252     362.138       36.114       2.8280E+05   2.5452E+02
    9       1       1      17   2.3600E+02    8.1645E+03     313.911     283.448       30.463       2.4872E+05   2.2385E+02
   10       1       1      16   2.9830E+02    8.6612E+03     243.711     200.605       43.106       3.7335E+05   3.3601E+02
 :
--------------------------------------------------------------------------------------------------------------
  HEAT FLUX BOUNDARY CONDITION AT END 2:          HEAT FLUX (W/sq.m) =   0.0000E+00   Q (W) =   0.0000E+00
--------------------------------------------------------------------------------------------------------------
  STRUCTURE TOTALS: INTERNAL POWER (W) =   2.5000E+03   EXCHANGE POWER (W) =   2.5000E+03   NET POWER (W) =  -2.2737E-12

 *TOTAL INTERNAL POWER FOR ALL COPIES OF ALL STRUCTURES (W) =   5.0000E+03
 *TOTAL EXCHANGE POWER FOR ALL COPIES OF ALL STRUCTURES (W) =   5.0000E+03
 *NET POWER (INTERNAL - EXCHANGE) FOR ALL COPIES OF ALL STRUCTURES (W) =  -1.8190E-12


v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.
       ---> SYSTEM TIME (s) =     30.000        TIME STEP #      3000    TIME STEP (s) =    1.0000E-02 <---

><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><

  *** STRUCTURE #    1    STRUCTURE NAME: BED1                     NUMBER OF COPIES =   1.0000E+00       DYNAMIC SOLUTION

  * TIME (s) =     30.000        SYSTEM TIME STEP #     3000   SYSTEM TIME STEP (s) =    1.0000E-02
  * HEAT TRANSFER TIME STEP #      3073   HEAT TRANSFER TIME STEP (s) =   1.0000E-02   EXPLICIT LIMIT (s) =   2.0816E-02

  HEAT FLUX BOUNDARY CONDITION AT END 1:          HEAT FLUX (W/sq.m) =   0.0000E+00   Q (W) =   0.0000E+00
--------------------------------------------------------------------------------------------------------------
 e/n#   I FAC    Tn (K)  dT/dt (K/s)    Te (K)  dT/dX (K/m)    POWER (W)    MATERIAL    COND (W/m*K)  C (J/kg*K)  DEN (kg/cub.m)
    1  0.5370   781.569 -6.9062E-01   765.445 -3.2248E+04  2.5000E+02  BED,D=400C    2.0000E+00   3.6000E+02    3.5580E+03
    2  0.5372   749.321 -6.5976E-01   717.472 -6.3698E+04  2.5000E+02  BED,D=400C    2.0000E+00   3.6000E+02    3.5580E+03
    3  0.5377   685.623 -5.9497E-01   654.027 -6.3192E+04  2.5000E+02  BED,D=400C    2.0000E+00   3.6000E+02    3.5580E+03
    4  0.5383   622.431 -5.2711E-01   590.823 -6.3215E+04  2.5000E+02  BED,D=400C    2.0000E+00   3.6000E+02    3.5580E+03
    5  0.5389   559.216 -4.5946E-01   527.572 -6.3287E+04  2.5000E+02  BED,D=400C    2.0000E+00   3.6000E+02    3.5580E+03
 :
    6  0.5397   495.929 -3.9188E-01   464.365 -6.3127E+04  2.5000E+02  BED,D=400C    2.0000E+00   3.6000E+02    3.5580E+03
    7  0.5406   432.802 -3.2453E-01   400.886 -6.3832E+04  2.5000E+02  BED,D=400C    2.0000E+00   3.6000E+02    3.5580E+03
    8  0.5416   368.970 -2.5677E-01   338.242 -6.1455E+04  2.5000E+02  BED,D=400C    2.0000E+00   3.6000E+02    3.5580E+03
    9  0.5430   307.514 -1.9126E-01   272.449 -7.0132E+04  2.5000E+02  BED,D=400C    2.0000E+00   3.6000E+02    3.5580E+03
   10  0.5447   237.383 -1.1733E-01   217.477 -3.9811E+04  2.5000E+02  BED,D=400C    2.0000E+00   3.6000E+02    3.5580E+03
 :
   11  0.5458   197.571 -7.6548E-02
--------------------------------------------------------------------------------------------------------------
   e#    SURFACE    Q'' (W/sq.m)       Q (W)
--------------------------------------------------------------------------------------------------------------
   e#    SURFACE  NW #     E#    REYNOLDS #  HTC (W/sq.m*K)   Twall (K)   Tfluid (K)  FIlMDROP (K)  Q'' (W/sq.m)     Q (W)
    1       1      1      15    1.5115E+02    7.5383E+03      765.448     735.862      29.586      2.2303E+05   2.0073E+02
    2       1      1      14    1.5929E+02    7.6380E+03      717.475     681.057      36.418      2.7816E+05   2.5035E+02
    3       1      1      13    1.7007E+02    7.7645E+03      654.030     618.342      35.688      2.7710E+05   2.4939E+02
    4       1      1      12    1.8282E+02    7.9067E+03      590.826     555.783      35.042      2.7707E+05   2.4936E+02
    5       1      1      11    1.9822E+02    8.0690E+03      527.574     493.273      34.301      2.7678E+05   2.4910E+02
 :
    6       1      1      10    2.1727E+02    8.2574E+03      464.367     430.798      33.569      2.7719E+05   2.4947E+02
    7       1      1       9    2.4154E+02    8.4806E+03      400.887     368.484      32.403      2.7480E+05   2.4732E+02
    8       1      1       8    2.7397E+02    8.7544E+03      338.243     305.980      32.263      2.8244E+05   2.5420E+02
    9       1      1       7    3.1880E+02    9.0963E+03      272.449     244.673      27.776      2.5266E+05   2.2739E+02
   10       1      1       6    3.9250E+02    9.5888E+03      217.477     180.027      37.451      3.5911E+05   3.2320E+02
 :
--------------------------------------------------------------------------------------------------------------
  HEAT FLUX BOUNDARY CONDITION AT END 2:          HEAT FLUX (W/sq.m) =   0.0000E+00   Q (W) =   0.0000E+00
--------------------------------------------------------------------------------------------------------------
  STRUCTURE TOTALS: INTERNAL POWER (W) =   2.5000E+03   EXCHANGE POWER (W) =   2.5005E+03   NET POWER (W) =  -5.0040E-01

><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><

  *** STRUCTURE #    2    STRUCTURE NAME: BED2                     NUMBER OF COPIES =   1.0000E+00       DYNAMIC SOLUTION

  * TIME (s) =     30.000        SYSTEM TIME STEP #     3000   SYSTEM TIME STEP (s) =    1.0000E-02
  * HEAT TRANSFER TIME STEP #      3000   HEAT TRANSFER TIME STEP (s) =   1.0000E-02   EXPLICIT LIMIT (s) =   3.7315E-02

  HEAT FLUX BOUNDARY CONDITION AT END 1:          HEAT FLUX (W/sq.m) =   0.0000E+00   Q (W) =   0.0000E+00
--------------------------------------------------------------------------------------------------------------
 e/n#   I FAC    Tn (K)  dT/dt (K/s)    Te (K)  dT/dX (K/m)    POWER (W)    MATERIAL    COND (W/m*K)  C (J/kg*K)  DEN (kg/cub.m)
    1  0.5203  1947.960  5.5102E+00  1909.643 -7.6635E+04  2.5000E+02  BED,D=400C    2.0000E+00   3.6000E+02    3.5580E+03
    2  0.5204  1871.326  5.3190E+00  1789.654 -1.6334E+05  2.5000E+02  BED,D=400C    2.0000E+00   3.6000E+02    3.5580E+03
    3  0.5207  1707.982  4.8722E+00  1620.344 -1.7527E+05  2.5000E+02  BED,D=400C    2.0000E+00   3.6000E+02    3.5580E+03
    4  0.5210  1532.707  4.3475E+00  1444.209 -1.7700E+05  2.5000E+02  BED,D=400C    2.0000E+00   3.6000E+02    3.5580E+03
    5  0.5214  1355.710  3.8074E+00  1267.040 -1.7734E+05  2.5000E+02  BED,D=400C    2.0000E+00   3.6000E+02    3.5580E+03
 :
    6  0.5218  1178.370  3.2660E+00  1089.816 -1.7711E+05  2.5000E+02  BED,D=400C    2.0000E+00   3.6000E+02    3.5580E+03
    7  0.5224  1001.262  2.7262E+00   912.107 -1.7831E+05  2.5000E+02  BED,D=400C    2.0000E+00   3.6000E+02    3.5580E+03
    8  0.5231   822.952  2.1872E+00   736.349 -1.7321E+05  2.5000E+02  BED,D=400C    2.0000E+00   3.6000E+02    3.5580E+03
    9  0.5239   649.746  1.6597E+00   552.661 -1.9417E+05  2.5000E+02  BED,D=400C    2.0000E+00   3.6000E+02    3.5580E+03
   10  0.5253   455.576  1.0849E+00   402.127 -1.0690E+05  2.5000E+02  BED,D=400C    2.0000E+00   3.6000E+02    3.5580E+03
 :
   11  0.5261   348.678  7.7875E-01
--------------------------------------------------------------------------------------------------------------
   e#    SURFACE    Q'' (W/sq.m)       Q (W)
--------------------------------------------------------------------------------------------------------------
   e#    SURFACE  NW #     E#    REYNOLDS #  HTC (W/sq.m*K)   Twall (K)   Tfluid (K)  FILMDROP (K)  Q'' (W/sq.m)     Q (W)
    1       1      1      25    3.0063E+01    3.7579E+03     1903.617    1857.744      51.873      1.9494E+05   1.7544E+02
    2       1      1      24    3.1674E+01    3.8069E+03     1789.629    1720.038      69.592      2.6493E+05   2.3844E+02
    3       1      1      23    3.3997E+01    3.8743E+03     1620.322    1549.497      70.825      2.7440E+05   2.4696E+02
    4       1      1      22    3.6875E+01    3.9532E+03     1444.189    1374.445      69.744      2.7571E+05   2.4814E+02
    5       1      1      21    4.0454E+01    4.0451E+03     1267.023    1198.877      68.147      2.7566E+05   2.4810E+02
 :
    6       1      1      20    4.5035E+01    4.1543E+03     1089.802    1023.365      66.437      2.7600E+05   2.4840E+02
    7       1      1      19    5.1141E+01    4.2876E+03      912.095     848.318      63.777      2.7345E+05   2.4611E+02
   8       1       1     18   5.9831E+01    4.4582E+03     736.340     672.980      63.360      2.8247E+05    2.5422E+02
   9       1       1     17   7.3045E+01    4.6851E+03     552.654     501.349      51.306      2.4037E+05    2.1634E+02
  10       1       1     16   9.9032E+01    5.0544E+03     402.123     319.971      82.152      4.1523E+05    3.7370E+02
 :
------------------------------------------------------------------------------------------------------------------------
   HEAT FLUX BOUNDARY CONDITION AT END 2:          HEAT FLUX (W/sq.m) =   0.0000E+00   Q (W) =   0.0000E+00
------------------------------------------------------------------------------------------------------------------------
   STRUCTURE TOTALS: INTERNAL POWER (W) =   2.5000E+03   EXCHANGE POWER (W) =   2.4958E+03   NET POWER (W) =   4.1519E+00

  *TOTAL INTERNAL POWER FOR ALL COPIES OF ALL STRUCTURES (W) =   5.0000E+03
  *TOTAL EXCHANGE POWER FOR ALL COPIES OF ALL STRUCTURES (W) =   4.9963E+03
  *NET POWER (INTERNAL - EXCHANGE) FOR ALL COPIES OF ALL STRUCTURES (W) =   3.6515E+00


v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.
      ---> SYSTEM TIME (s) =    60.000       TIME STEP #     6000   TIME STEP (s) =   1.0000E-02 <---

><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><

   *** STRUCTURE #    1   STRUCTURE NAME: BED1                     NUMBER OF COPIES =   1.0000E+00      DYNAMIC SOLUTION

   * TIME (s) =    60.000       SYSTEM TIME STEP #     6000   SYSTEM TIME STEP (s) =   1.0000E-02
   * HEAT TRANSFER TIME STEP #     6073   HEAT TRANSFER TIME STEP (s) =   1.0000E-02   EXPLICIT LIMIT (s) =   2.0473E-02

   HEAT FLUX BOUNDARY CONDITION AT END 1:          HEAT FLUX (W/sq.m) =   0.0000E+00   Q (W) =   0.0000E+00
------------------------------------------------------------------------------------------------------------------------
  e/n#   I FAC    Tn (K)  dT/dt (K/s)    Te (K)  dT/dX (K/m)    POWER (W)    MATERIAL   COND (W/m*K)  C (J/kg*K)  DEN (kg/cub.m)
     1  0.5376    764.094 -7.1853E-01    748.357 -3.1473E+04  2.5000E+02  BED,D=400C    2.0000E+00   3.6000E+02    3.5580E+03
     2  0.5379    732.621 -6.8688E-01    701.586 -6.2070E+04  2.5000E+02  BED,D=400C    2.0000E+00   3.6000E+02    3.5580E+03
     3  0.5384    670.551 -6.2033E-01    639.810 -6.1484E+04  2.5000E+02  BED,D=400C    2.0000E+00   3.6000E+02    3.5580E+03
     4  0.5389    609.068 -5.5041E-01    578.313 -6.1509E+04  2.5000E+02  BED,D=400C    2.0000E+00   3.6000E+02    3.5580E+03
     5  0.5396    547.558 -4.8051E-01    516.768 -6.1580E+04  2.5000E+02  BED,D=400C    2.0000E+00   3.6000E+02    3.5580E+03
  :
     6  0.5404    485.978 -4.1045E-01    455.266 -6.1423E+04  2.5000E+02  BED,D=400C    2.0000E+00   3.6000E+02    3.5580E+03
     7  0.5413    424.555 -3.4040E-01    393.497 -6.2115E+04  2.5000E+02  BED,D=400C    2.0000E+00   3.6000E+02    3.5580E+03
     8  0.5424    362.440 -2.6968E-01    332.544 -5.9792E+04  2.5000E+02  BED,D=400C    2.0000E+00   3.6000E+02    3.5580E+03
     9  0.5437    302.648 -2.0110E-01    268.522 -6.8251E+04  2.5000E+02  BED,D=400C    2.0000E+00   3.6000E+02    3.5580E+03
    10  0.5455    234.397 -1.2337E-01    215.011 -3.8772E+04  2.5000E+02  BED,D=400C    2.0000E+00   3.6000E+02    3.5580E+03
  :
    11  0.5465    195.625 -8.0368E-02
------------------------------------------------------------------------------------------------------------------------
   e#    SURFACE   Q'' (W/sq.m)       Q (W)
------------------------------------------------------------------------------------------------------------------------
   e#    SURFACE  NW #      E#    REYNOLDS #  HTC (W/sq.m*K)   Twall (K)   Tfluid (K)  FILMDROP (K)  Q'' (W/sq.m)      Q (W)
    1       1       1      15    1.5777E+02     7.6898E+03      748.361     719.289       29.072       2.2356E+05    2.0120E+02
    2       1       1      14    1.6624E+02     7.7913E+03      701.589     665.866       35.723       2.7833E+05    2.5050E+02
    3       1       1      13    1.7744E+02     7.9198E+03      639.812     604.823       34.989       2.7710E+05    2.4939E+02
    4       1       1      12    1.9067E+02     8.0641E+03      578.315     543.955       34.360       2.7708E+05    2.4937E+02
    5       1       1      11    2.0662E+02     8.2286E+03      516.770     483.132       33.638       2.7679E+05    2.4911E+02
  :
    6       1       1      10    2.2634E+02     8.4196E+03      455.268     422.343       32.924       2.7721E+05    2.4949E+02
    7       1       1       9    2.5141E+02     8.6454E+03      393.499     361.710       31.789       2.7483E+05    2.4735E+02
    8       1       1       8    2.8482E+02     8.9220E+03      332.545     300.890       31.655       2.8243E+05    2.5418E+02
    9       1       1       7    3.3084E+02     9.2665E+03      268.523     241.228       27.295       2.5293E+05    2.2764E+02
   10       1       1       6    4.0603E+02     9.7606E+03      215.012     178.324       36.688       3.5810E+05    3.2229E+02
  :
------------------------------------------------------------------------------------------------------------------------
   HEAT FLUX BOUNDARY CONDITION AT END 2:          HEAT FLUX (W/sq.m) =   0.0000E+00   Q (W) =   0.0000E+00
------------------------------------------------------------------------------------------------------------------------
   STRUCTURE TOTALS: INTERNAL POWER (W) =   2.5000E+03   EXCHANGE POWER (W) =   2.5005E+03   NET POWER (W) =  -5.2293E-01

><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><

   *** STRUCTURE #    2   STRUCTURE NAME: BED2                     NUMBER OF COPIES =   1.0000E+00      DYNAMIC SOLUTION

   * TIME (s) =    60.000       SYSTEM TIME STEP #     6000   SYSTEM TIME STEP (s) =   1.0000E-02
   * HEAT TRANSFER TIME STEP #     6000   HEAT TRANSFER TIME STEP (s) =   1.0000E-02   EXPLICIT LIMIT (s) =   3.9278E-02

   HEAT FLUX BOUNDARY CONDITION AT END 1:          HEAT FLUX (W/sq.m) =   0.0000E+00   Q (W) =   0.0000E+00
------------------------------------------------------------------------------------------------------------------------
  e/n#   I FAC    Tn (K)  dT/dt (K/s)    Te (K)  dT/dX (K/m)    POWER (W)    MATERIAL   COND (W/m*K)  C (J/kg*K)  DEN (kg/cub.m)
     1  0.5193  2100.788  6.8815E+00  2059.933 -8.1710E+04  2.5000E+02  BED,D=400C    2.0000E+00   3.6000E+02    3.5580E+03
     2  0.5194  2019.078  6.6631E+00  1931.407 -1.7534E+05  2.5000E+02  BED,D=400C    2.0000E+00   3.6000E+02    3.5580E+03
     3  0.5197  1843.735  6.1403E+00  1748.956 -1.8956E+05  2.5000E+02  BED,D=400C    2.0000E+00   3.6000E+02    3.5580E+03
     4  0.5200  1654.177  5.5093E+00  1558.271 -1.9181E+05  2.5000E+02  BED,D=400C    2.0000E+00   3.6000E+02    3.5580E+03
     5  0.5203  1462.364  4.8498E+00  1366.230 -1.9227E+05  2.5000E+02  BED,D=400C    2.0000E+00   3.6000E+02    3.5580E+03
  :
     6  0.5208  1270.095  4.1817E+00  1174.064 -1.9206E+05  2.5000E+02  BED,D=400C    2.0000E+00   3.6000E+02    3.5580E+03
     7  0.5213  1078.034  3.5094E+00   981.383 -1.9330E+05  2.5000E+02  BED,D=400C    2.0000E+00   3.6000E+02    3.5580E+03
     8  0.5219   884.732  2.8325E+00   790.755 -1.8795E+05  2.5000E+02  BED,D=400C    2.0000E+00   3.6000E+02    3.5580E+03
     9  0.5228   696.778  2.1633E+00   591.633 -2.1029E+05  2.5000E+02  BED,D=400C    2.0000E+00   3.6000E+02    3.5580E+03
    10  0.5240   486.489  1.1295E+00   428.742 -1.1549E+05  2.5000E+02  BED,D=400C    2.0000E+00   3.6000E+02    3.5580E+03
  :
    11  0.5248   370.994  1.0377E+00
------------------------------------------------------------------------------------------------------------------------
   e#    SURFACE   Q'' (W/sq.m)       Q (W)
------------------------------------------------------------------------------------------------------------------------
   e#    SURFACE  NW #      E#    REYNOLDS #  HTC (W/sq.m*K)   Twall (K)   Tfluid (K)  FILMDROP (K)  Q'' (W/sq.m)      Q (W)
    1       1       1      25    2.6299E+01     3.5375E+03     2059.900    2005.537       54.364       1.9231E+05    1.7308E+02
    2       1       1      24    2.7697E+01     3.5832E+03     1931.376    1857.923       73.453       2.6320E+05    2.3688E+02
    3       1       1      23    2.9727E+01     3.6466E+03     1748.928    1673.843       75.085       2.7380E+05    2.4642E+02
    4       1       1      22    3.2251E+01     3.7210E+03     1558.246    1484.224       74.022       2.7543E+05    2.4789E+02
    5       1       1      21    3.5395E+01     3.8079E+03     1366.208    1293.866       72.343       2.7547E+05    2.4792E+02
 :
  6      1      1     20   3.9426E+01    3.9111E+03    1174.046    1103.526      70.520       2.7581E+05    2.4823E+02
  7      1      1     19   4.4808E+01    4.0374E+03     981.368     913.671      67.697       2.7332E+05    2.4599E+02
  8      1      1     18   5.2487E+01    4.1992E+03     790.743     723.507      67.237       2.8234E+05    2.5410E+02
  9      1      1     17   6.4208E+01    4.4150E+03     591.625     537.383      54.242       2.3948E+05    2.1553E+02
 10      1      1     16   8.7479E+01    4.7684E+03     428.736     340.500      88.236       4.2075E+05    3.7867E+02
 :
---------------------------------------------------------------------------------------------------------------------
  HEAT FLUX BOUNDARY CONDITION AT END 2:          HEAT FLUX (W/sq.m) =   0.0000E+00   Q (W) =   0.0000E+00
---------------------------------------------------------------------------------------------------------------------
  STRUCTURE TOTALS: INTERNAL POWER (W) =   2.5000E+03   EXCHANGE POWER (W) =   2.4947E+03    NET POWER (W) =   5.2822E+00

 *TOTAL INTERNAL POWER FOR ALL COPIES OF ALL STRUCTURES (W) =   5.0000E+03
 *TOTAL EXCHANGE POWER FOR ALL COPIES OF ALL STRUCTURES (W) =   4.9952E+03
 *NET POWER (INTERNAL - EXCHANGE) FOR ALL COPIES OF ALL STRUCTURES (W) =   4.7592E+00


><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><

  *** STRUCTURE #    1   STRUCTURE NAME: BED1                         NUMBER OF COPIES =   1.0000E+00       DYNAMIC SOLUTION

  * TIME (s) =    88.440        SYSTEM TIME STEP #     8844   SYSTEM TIME STEP (s) =   1.0000E-02
  * HEAT TRANSFER TIME STEP #    10000   HEAT TRANSFER TIME STEP (s) =   5.0000E-03   EXPLICIT LIMIT (s) =   1.8621E-02

  HEAT FLUX BOUNDARY CONDITION AT END 1:          HEAT FLUX (W/sq.m) =   0.0000E+00   Q (W) =   0.0000E+00
---------------------------------------------------------------------------------------------------------------------
 e/n#   I FAC    Tn (K)  dT/dt (K/s)    Te (K)  dT/dX (K/m)    POWER (W)    MATERIAL   COND (W/m*K)  C (J/kg*K)  DEN (kg/cub.m)
    1  0.5213   674.864 -9.2161E+00   661.097 -2.7535E+04  2.5000E+02  BED,D=400C    2.0000E+00   3.6000E+02    3.5580E+03
    2  0.5214   647.330 -8.7742E+00   620.423 -5.3815E+04  2.5000E+02  BED,D=400C    2.0000E+00   3.6000E+02    3.5580E+03
    3  0.5217   593.515 -7.8678E+00   567.099 -5.2833E+04  2.5000E+02  BED,D=400C    2.0000E+00   3.6000E+02    3.5580E+03
    4  0.5220   540.682 -6.9406E+00   514.260 -5.2844E+04  2.5000E+02  BED,D=400C    2.0000E+00   3.6000E+02    3.5580E+03
    5  0.5224   487.838 -6.0191E+00   461.401 -5.2874E+04  2.5000E+02  BED,D=400C    2.0000E+00   3.6000E+02    3.5580E+03
 :
    6  0.5228   434.964 -5.0995E+00   408.614 -5.2698E+04  2.5000E+02  BED,D=400C    2.0000E+00   3.6000E+02    3.5580E+03
    7  0.5233   382.265 -4.1855E+00   355.626 -5.3279E+04  2.5000E+02  BED,D=400C    2.0000E+00   3.6000E+02    3.5580E+03
    8  0.5239   328.987 -3.2668E+00   303.380 -5.1212E+04  2.5000E+02  BED,D=400C    2.0000E+00   3.6000E+02    3.5580E+03
    9  0.5246   277.774 -2.3900E+00   248.534 -5.8481E+04  2.5000E+02  BED,D=400C    2.0000E+00   3.6000E+02    3.5580E+03
   10  0.5256   219.294 -1.3832E+00   202.621 -3.3345E+04  2.5000E+02  BED,D=400C    2.0000E+00   3.6000E+02    3.5580E+03
 :
   11  0.5261   185.948 -8.1733E-01
---------------------------------------------------------------------------------------------------------------------
   e#    SURFACE    Q'' (W/sq.m)       Q (W)
---------------------------------------------------------------------------------------------------------------------
   e#    SURFACE  NW #     E#   REYNOLDS #  HTC (W/sq.m*K)  Twall (K)   Tfluid (K)   FILMDROP (K)  Q'' (W/sq.m)      Q (W)
    1       1       1     15   2.0106E+02    8.6134E+03     661.119      634.727       26.392       2.2732E+05    2.0459E+02
    2       1       1     14   2.1168E+02    8.7256E+03     620.442      588.317       32.126       2.8032E+05    2.5228E+02
    3       1       1     13   2.2554E+02    8.8661E+03     567.116      535.749       31.368       2.7811E+05    2.5030E+02
    4       1       1     12   2.4180E+02    9.0229E+03     514.275      483.462       30.813       2.7802E+05    2.5022E+02
    5       1       1     11   2.6128E+02    9.2009E+03     461.414      431.241       30.173       2.7762E+05    2.4986E+02
 :
    6       1       1     10   2.8514E+02    9.4061E+03     408.626      379.078       29.548       2.7793E+05    2.5014E+02
    7       1       1      9   3.1513E+02    9.6470E+03     355.635      327.076       28.559       2.7551E+05    2.4796E+02
    8       1       1      8   3.5449E+02    9.9388E+03     303.387      274.941       28.447       2.8273E+05    2.5445E+02
    9       1       1      7   4.0755E+02    1.0297E+04     248.539      223.799       24.740       2.5475E+05    2.2927E+02
   10       1       1      6   4.9112E+02    1.0798E+04     202.624      169.958       32.666       3.5273E+05    3.1746E+02
 :
---------------------------------------------------------------------------------------------------------------------
  HEAT FLUX BOUNDARY CONDITION AT END 2:          HEAT FLUX (W/sq.m) =   0.0000E+00   Q (W) =   0.0000E+00
---------------------------------------------------------------------------------------------------------------------
  STRUCTURE TOTALS: INTERNAL POWER (W) =   2.5000E+03   EXCHANGE POWER (W) =   2.5065E+03    NET POWER (W) =  -6.5253E+00

v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.
        ---> SYSTEM TIME (s) =    90.000        TIME STEP #     9000   TIME STEP (s) =   1.0000E-02 <---

><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><

  *** STRUCTURE #    1   STRUCTURE NAME: BED1                         NUMBER OF COPIES =   1.0000E+00       DYNAMIC SOLUTION

  * TIME (s) =    90.000        SYSTEM TIME STEP #     9000   SYSTEM TIME STEP (s) =   1.0000E-02
  * HEAT TRANSFER TIME STEP #    10312   HEAT TRANSFER TIME STEP (s) =   5.0000E-03   EXPLICIT LIMIT (s) =   1.8271E-02

  HEAT FLUX BOUNDARY CONDITION AT END 1:          HEAT FLUX (W/sq.m) =   0.0000E+00   Q (W) =   0.0000E+00
---------------------------------------------------------------------------------------------------------------------
 e/n#   I FAC    Tn (K)  dT/dt (K/s)    Te (K)  dT/dX (K/m)    POWER (W)    MATERIAL   COND (W/m*K)  C (J/kg*K)  DEN (kg/cub.m)
    1  0.5217   658.091 -8.7631E+00   644.717 -2.6748E+04  2.5000E+02  BED,D=400C    2.0000E+00   3.6000E+02    3.5580E+03
    2  0.5218   631.343 -8.3246E+00   605.248 -5.2189E+04  2.5000E+02  BED,D=400C    2.0000E+00   3.6000E+02    3.5580E+03
    3  0.5221   579.154 -7.4323E+00   553.576 -5.1156E+04  2.5000E+02  BED,D=400C    2.0000E+00   3.6000E+02    3.5580E+03
    4  0.5225   527.998 -6.5280E+00   502.411 -5.1173E+04  2.5000E+02  BED,D=400C    2.0000E+00   3.6000E+02    3.5580E+03
    5  0.5228   476.825 -5.6329E+00   451.223 -5.1203E+04  2.5000E+02  BED,D=400C    2.0000E+00   3.6000E+02    3.5580E+03
 :
    6  0.5233   425.622 -4.7433E+00   400.106 -5.1032E+04  2.5000E+02  BED,D=400C    2.0000E+00   3.6000E+02    3.5580E+03
    7  0.5238   374.589 -3.8630E+00   348.789 -5.1600E+04  2.5000E+02  BED,D=400C    2.0000E+00   3.6000E+02    3.5580E+03
    8  0.5244   322.990 -2.9817E+00   298.194 -4.9592E+04  2.5000E+02  BED,D=400C    2.0000E+00   3.6000E+02    3.5580E+03
    9  0.5251   273.398 -2.1448E+00   245.076 -5.6643E+04  2.5000E+02  BED,D=400C    2.0000E+00   3.6000E+02    3.5580E+03
   10  0.5261   216.755 -1.1875E+00   200.592 -3.2325E+04  2.5000E+02  BED,D=400C    2.0000E+00   3.6000E+02    3.5580E+03
 :
   11  0.5266   184.430 -6.4999E-01
---------------------------------------------------------------------------------------------------------------------
   e#    SURFACE    Q'' (W/sq.m)       Q (W)
---------------------------------------------------------------------------------------------------------------------
   e#    SURFACE  NW #     E#   REYNOLDS #  HTC (W/sq.m*K)  Twall (K)   Tfluid (K)   FILMDROP (K)  Q'' (W/sq.m)      Q (W)
    1       1       1     15   2.1117E+02    8.8137E+03     644.738      618.887       25.851       2.2784E+05    2.0506E+02
    2       1       1     14   2.2227E+02    8.9280E+03     605.267      573.860       31.408       2.8041E+05    2.5237E+02
   3      1       1     13  2.3672E+02     9.0707E+03      553.593      522.940        30.653        2.7804E+05     2.5024E+02
   4      1       1     12  2.5363E+02     9.2299E+03      502.426      472.309        30.117        2.7798E+05     2.5018E+02
   5      1       1     11  2.7387E+02     9.4104E+03      451.236      421.738        29.497        2.7758E+05     2.4982E+02
 :
   6      1       1     10  2.9860E+02     9.6183E+03      400.116      371.223        28.893        2.7790E+05     2.5011E+02
   7      1       1      9  3.2961E+02     9.8618E+03      348.798      320.861        27.936        2.7550E+05     2.4795E+02
   8      1       1      8  3.7018E+02     1.0156E+04      298.200      270.368        27.832        2.8266E+05     2.5440E+02
   9      1       1      7  4.2460E+02     1.0516E+04      245.080      220.829        24.251        2.5503E+05     2.2953E+02
  10      1       1      6  5.0965E+02     1.1017E+04      200.595      168.682        31.913        3.5159E+05     3.1644E+02
 :
-----------------------------------------------------------------------------------------------------------------------------------
   HEAT FLUX BOUNDARY CONDITION AT END 2:          HEAT FLUX (W/sq.m) =   0.0000E+00   Q (W) =   0.0000E+00
-----------------------------------------------------------------------------------------------------------------------------------
   STRUCTURE TOTALS: INTERNAL POWER (W) =   2.5000E+03   EXCHANGE POWER (W) =   2.5061E+03   NET POWER (W) =  -6.0899E+00

><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><

   *** STRUCTURE #    2   STRUCTURE NAME: BED2                     NUMBER OF COPIES =   1.0000E+00       DYNAMIC SOLUTION

   * TIME (s) =     90.000       SYSTEM TIME STEP #      9000   SYSTEM TIME STEP (s) =   1.0000E-02
   * HEAT TRANSFER TIME STEP #      9000   HEAT TRANSFER TIME STEP (s) =   1.0000E-02   EXPLICIT LIMIT (s) =   7.4820E-02

   HEAT FLUX BOUNDARY CONDITION AT END 1:          HEAT FLUX (W/sq.m) =   0.0000E+00   Q (W) =   0.0000E+00
-----------------------------------------------------------------------------------------------------------------------------------
  e/n#    I FAC    Tn (K)  dT/dt (K/s)    To (K)  dT/dX (K/m)     POWER (W)     MATERIAL    COND (W/m*K)  C (J/kg*K)  DEN (kg/cub.m)
     1  0.5108  4485.683  5.8520E+02  4425.767 -1.1983E+05  2.5000E+02  BED,D=400C      2.0000E+00    3.6000E+02      3.5580E+03
     2  0.5108  4365.851  5.8170E+02  4223.478 -2.8475E+05  2.5000E+02  BED,D=400C      2.0000E+00    3.6000E+02      3.5580E+03
     3  0.5109  4081.105  5.6729E+02  3906.656 -3.4890E+05  2.5000E+02  BED,D=400C      2.0000E+00    3.6000E+02      3.5580E+03
     4  0.5111  3732.207  5.4018E+02  3543.292 -3.7783E+05  2.5000E+02  BED,D=400C      2.0000E+00    3.6000E+02      3.5580E+03
     5  0.5112  3354.377  5.0411E+02  3156.763 -3.9523E+05  2.5000E+02  BED,D=400C      2.0000E+00    3.6000E+02      3.5580E+03
   :
     6  0.5114  2959.150  4.6154E+02  2754.546 -4.0921E+05  2.5000E+02  BED,D=400C      2.0000E+00    3.6000E+02      3.5580E+03
     7  0.5116  2549.941  4.1337E+02  2337.807 -4.2427E+05  2.5000E+02  BED,D=400C      2.0000E+00    3.6000E+02      3.5580E+03
     8  0.5119  2125.672  3.5976E+02  1909.649 -4.3205E+05  2.5000E+02  BED,D=400C      2.0000E+00    3.6000E+02      3.5580E+03
     9  0.5122  1693.626  3.0009E+02  1449.471 -4.8831E+05  2.5000E+02  BED,D=400C      2.0000E+00    3.6000E+02      3.5580E+03
    10  0.5128  1205.317  2.3070E+02  1072.345 -2.6594E+05  2.5000E+02  BED,D=400C      2.0000E+00    3.6000E+02      3.5580E+03
   :
    11  0.5132   939.374  1.9297E+02
-----------------------------------------------------------------------------------------------------------------------------------
    e#    SURFACE    Q'' (W/sq.m)       Q (W)
-----------------------------------------------------------------------------------------------------------------------------------
    e#    SURFACE  NW #       E#    REYNOLDS #  HTC (W/sq.m*K)  Twall (K)  Tfluid (K)  FILMDROP (K)  Q'' (W/sq.m)      Q (W)
     1       1      1       25    5.3135E+00    1.6654E+03    4422.913    4358.072      64.841      1.0798E+05    9.7186E+01
     2       1      1       24    5.5160E+00    1.6808E+03    4220.668    4124.148      96.520      1.6223E+05    1.4601E+02
     3       1      1       23    5.8349E+00    1.7043E+03    3903.948    3795.963     107.985      1.8404E+05    1.6564E+02
     4       1      1       22    6.2520E+00    1.7337E+03    3540.739    3428.341     112.399      1.9486E+05    1.7538E+02
     5       1      1       21    6.7826E+00    1.7689E+03    3154.404    3040.127     114.277      2.0215E+05    1.8193E+02
   :
     6       1      1       20    7.4688E+00    1.8116E+03    2752.409    2637.144     115.265      2.0881E+05    1.8793E+02
     7       1      1       19    8.3907E+00    1.8645E+03    2335.919    2221.073     114.846      2.1413E+05    1.9271E+02
     8       1      1       18    9.7122E+00    1.9331E+03    1908.039    1789.972     118.067      2.2824E+05    2.0541E+02
     9       1      1       17    1.1736E+01    2.0258E+03    1448.177    1353.759      94.418      1.9127E+05    1.7214E+02
    10       1      1       16    1.5982E+01    2.1866E+03    1071.314     858.526     212.787      4.6528E+05    4.1875E+02
   :
-----------------------------------------------------------------------------------------------------------------------------------
    HEAT FLUX BOUNDARY CONDITION AT END 2:          HEAT FLUX (W/sq.m) =   0.0000E+00   Q (W) =   0.0000E+00
-----------------------------------------------------------------------------------------------------------------------------------
    STRUCTURE TOTALS: INTERNAL POWER (W) =   2.5000E+03   EXCHANGE POWER (W) =   1.9431E+03   NET POWER (W) =   5.5690E+02

   *TOTAL INTERNAL POWER FOR ALL COPIES OF ALL STRUCTURES (W) =   5.0000E+03
   *TOTAL EXCHANGE POWER FOR ALL COPIES OF ALL STRUCTURES (W) =   4.4492E+03
   *NET POWER (INTERNAL - EXCHANGE) FOR ALL COPIES OF ALL STRUCTURES (W) =   5.5081E+02
```

PROGRAM SAFSIM OUTPUT (FUNCTION-CONTROLLED VARIABLES AND FUNCTIONS):

```
---------------------------------------------------------------------------------------------------------
EXAMPLE PROBLEM - INVESTIGATE STABILITY OF FLOW IN PARALLEL FLOW PATHS
CONTAINING POROUS BEDS WITH INTERNAL HEAT GENERATION
---------------------------------------------------------------------------------------------------------

v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.
      ---> SYSTEM TIME (s) =   0.00000      TIME STEP #       0   TIME STEP (s) =   1.0000E-02 <---

><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><

 *** TIME (s) =   0.00000        SYSTEM TIME STEP #       0   ***

 FUNCTION-CONTROLLED VARIABLES FOR FLUID MECHANICS:
   COUNTER   ID #    ............NAME...........      ARGUMENT F #      VALUE

 *** TIME (s) =   0.00000        SYSTEM TIME STEP #       0   ***

 FUNCTIONS FOR FLUID MECHANICS:
   COUNTER    F #    ID #    ............NAME............      VALUE       new-old VALUE  TIME DERIVATIVE
      1        1       8     P at BED INLET                  7.5791E+06     2.4390E-02     2.4390E+00
      2        2       8     P at BED1 OUTLET                7.5778E+06     0.0000E+00     0.0000E+00
      3        3     132     Pin - Pout OF BED1              1.3146E+03     2.4390E-02     2.4390E+00
      4        6       9     Tgas at BED1 INLET              1.2519E+02     6.6294E-06     6.6294E-04
      5        7       9     Tgas at BED1 OUTLET             6.9224E+02    -5.3058E-02    -5.3058E+00
      6        8     132     Tout - Tin OF BED1              5.6705E+02    -5.3064E-02    -5.3064E+00
      7        9     134     phi=(Tout-Tin)/Tin OF BED1      4.5296E+00    -4.2412E-04    -4.2412E-02
      8       17       9     Tgas at BED2 OUTLET             9.3324E+02     1.1745E-01     1.1745E+01
      9       18     132     Tout - Tin OF BED2              8.0805E+02     1.1744E-01     1.1744E+01
     10       19     134     phi=(Tout-Tin)/Tin OF BED2      6.4547E+00     9.3777E-04     9.3777E-02

><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><

 *** TIME (s) =   0.00000        SYSTEM TIME STEP #       0   ***

 FUNCTION-CONTROLLED VARIABLES FOR HEAT TRANSFER:
   COUNTER   ID #    ............NAME...........      ARGUMENT F #      VALUE

 *** TIME (s) =   0.00000        SYSTEM TIME STEP #       0   ***

 FUNCTIONS FOR HEAT TRANSFER:
   COUNTER    F #    ID #    ............NAME............      VALUE       new-old VALUE  TIME DERIVATIVE

><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><

 *** TIME (s) =   0.00000        SYSTEM TIME STEP #       0   ***

 FUNCTION-CONTROLLED VARIABLES FOR THE SYSTEM:
   COUNTER   ID #    ............NAME...........      ARGUMENT F #      VALUE
      1       35     MASS FLOW RATE BC AT NODE 1            10        5.0000E-04

 *** TIME (s) =   0.00000        SYSTEM TIME STEP #       0   ***

 FUNCTIONS FOR THE SYSTEM:
   COUNTER    F #    ID #    ............NAME............      VALUE       new-old VALUE  TIME DERIVATIVE
     11        5       1     PROBLEM TIME                    0.0000E+00     0.0000E+00     0.0000E+00
     12       10     114     TABLE, FLOWRATE BC VS TIME      5.0000E-04    -5.0000E-04    -5.0000E-02

v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.
      ---> SYSTEM TIME (s) =    30.000      TIME STEP #    3000   TIME STEP (s) =   1.0000E-02 <---

><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><

 *** TIME (s) =    30.000        SYSTEM TIME STEP #    3000   ***

 FUNCTION-CONTROLLED VARIABLES FOR FLUID MECHANICS:
   COUNTER   ID #    ............NAME...........      ARGUMENT F #      VALUE
                                                                          .
 *** TIME (s) =    30.000        SYSTEM TIME STEP #    3000   ***

 FUNCTIONS FOR FLUID MECHANICS:
   COUNTER    F #    ID #    ............NAME............      VALUE       new-old VALUE  TIME DERIVATIVE
      1        1       8     P at BED INLET                  7.5790E+06     8.1795E-03     8.1795E-01
      2        2       8     P at BED1 OUTLET                7.5778E+06     0.0000E+00     0.0000E+00
      3        3     132     Pin - Pout OF BED1              1.2262E+03     8.1795E-03     8.1795E-01
      4        6       9     Tgas at BED1 INLET              1.2549E+02     1.1603E-05     1.1603E-03
      5        7       9     Tgas at BED1 OUTLET             7.5407E+02    -6.7092E-03    -6.7092E-01
      6        8     132     Tout - Tin OF BED1              6.2858E+02    -6.7208E-03    -6.7208E-01
      7        9     134     phi=(Tout-Tin)/Tin OF BED1      5.0088E+00    -5.4018E-05    -5.4018E-03
      8       17       9     Tgas at BED2 OUTLET             1.8966E+03     5.4067E-02     5.4067E+00
      9       18     132     Tout - Tin OF BED2              1.7711E+03     5.4056E-02     5.4056E+00
     10       19     134     phi=(Tout-Tin)/Tin OF BED2      1.4113E+01     4.2944E-04     4.2944E-02

><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><

 *** TIME (s) =    30.000        SYSTEM TIME STEP #    3000   ***

 FUNCTION-CONTROLLED VARIABLES FOR HEAT TRANSFER:
   COUNTER   ID #    ............NAME...........      ARGUMENT F #      VALUE

 *** TIME (s) =    30.000        SYSTEM TIME STEP #    3000   ***
 FUNCTIONS FOR HEAT TRANSFER:
   COUNTER    F #    ID #     ............NAME............         VALUE       new-old VALUE  TIME DERIVATIVE

><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><

 *** TIME (s) =    30.000         SYSTEM TIME STEP #    3000    ***

 FUNCTION-CONTROLLED VARIABLES FOR THE SYSTEM:
   COUNTER   ID #     ............NAME............       ARGUMENT F #       VALUE
       1       35    MASS FLOW RATE BC AT NODE 1            10         3.5900E-04

 *** TIME (s) =    30.000         SYSTEM TIME STEP #    3000    ***

 FUNCTIONS FOR THE SYSTEM:
   COUNTER    F #    ID #     ............NAME............         VALUE       new-old VALUE  TIME DERIVATIVE
      11       5       1     PROBLEM TIME                       3.0000E+01    1.0000E-02     1.0000E+00
      12      10     114     TABLE, FLOWRATE BC VS TIME         3.5900E-04    0.0000E+00     0.0000E+00

v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.
       ---> SYSTEM TIME (s) =    60.000        TIME STEP #     6000   TIME STEP (s) =   1.0000E-02 <---

><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><

 *** TIME (s) =    60.000         SYSTEM TIME STEP #    6000    ***

 FUNCTION-CONTROLLED VARIABLES FOR FLUID MECHANICS:
   COUNTER   ID #     ............NAME............       ARGUMENT F #       VAIUE

 *** TIME (s) =    60.000         SYSTEM TIME STEP #    6000    ***

 FUNCTIONS FOR FLUID MECHANICS:
   COUNTER    F #    ID #     ............NAME............         VALUE       new-old VALUE  TIME DERIVATIVE
       1       1       8     P at BED INLET                     7.5790E+06    9.7730E-03     9.7730E-01
       2       2       8     P at BED1 OUTLET                   7.5778E+06    0.0000E+00     0.0000E+00
       3       3     132     Pin - Pout OF BED1                 1.2483E+03    9.7730E-03     9.7730E-01
       4       6       9     Tgas at BED1 INLET                 1.2553E+02    1.5749E-05     1.5749E-03
       5       7       9     Tgas at BED1 OUTLET                7.3709E+02   -6.9798E-03    -6.9798E-01
       6       8     132     Tout - Tin OF BED1                 6.1157E+02   -6.9955E-03    -6.9955E-01
       7       9     134     phi=(Tout-Tin)/Tin OF BED1         4.8720E+00   -5.6340E-05    -5.6340E-03
       8      17       9     Tgas at BED2 OUTLET                2.0466E+03    6.7631E-02     6.7631E+00
       9      18     132     Tout - Tin OF BED2                 1.9211E+03    6.7615E-02     6.7615E+00
      10      19     134     phi=(Tout-Tin)/Tin OF BED2         1.5304E+01    5.3672E-04     5.3672E-02

><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><

 *** TIME (s) =    60.000         SYSTEM TIME STEP #    6000    ***

 FUNCTION-CONTROLLED VARIABLES FOR HEAT TRANSFER:
   COUNTER   ID #     ............NAME............       ARGUMENT F #       VALUE

 *** TIME (s) =    60.000         SYSTEM TIME STEP #    6000    ***

 FUNCTIONS FOR HEAT TRANSFER:
   COUNTER    F #    ID #     ............NAME............         VALUE       new-old VALUE  TIME DERIVATIVE

><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><

 *** TIME (s) =    60.000         SYSTEM TIME STEP #    6000    ***

 FUNCTION-CONTROLLED VARIABLES FOR THE SYSTEM:
   COUNTER   ID #     ............NAME............       ARGUMENT F #       VALUE
       1       35    MASS FLOW RATE BC AT NODE 1            10         3.5900E-04

 *** TIME (s) =    60.000         SYSTEM TIME STEP #    6000    ***

 FUNCTIONS FOR THE SYSTEM:
   COUNTER    F #    ID #     ............NAME............         VALUE       new-old VALUE  TIME DERIVATIVE
      11       5       1     PROBLEM TIME                       6.0000E+01    1.0000E-02     1.0000E+00
      12      10     114     TABLE, FLOWRATE BC VS TIME         3.5900E-04    0.0000E+00     0.0000E+00

v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.v.
       ---> SYSTEM TIME (s) =    90.000        TIME STEP #     9000   TIME STEP (s) =   1.0000E-02 <---

><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><

 *** TIME (s) =    90.000         SYSTEM TIME STEP #    9000    ***

 FUNCTION-CONTROLLED VARIABLES FOR FLUID MECHANICS:
   COUNTER   ID #     ............NAME............       ARGUMENT F #       VALUE

 *** TIME (s) =    90.000         SYSTEM TIME STEP #    9000    ***

 FUNCTIONS FOR FLUID MECHANICS:
   COUNTER    F #    ID #     ............NAME............         VALUE       new-old VALUE  TIME DERIVATIVE
       1       1       8     P at BED INLET                     7.5792E+06    2.4081E-01     2.4081E+01
       2       2       8     P at BED1 OUTLET                   7.5778E+06    0.0000E+00     0.0000E+00
       3       3     132     Pin - Pout OF BED1                 1.4427E+03    2.4081E-01     2.4081E+01
       4       6       9     Tgas at BED1 INLET                 1.2641E+02    2.9498E-03     2.9498E-01
       5       7       9     Tgas at BED1 OUTLET                6.3422E+02   -1.0007E-01    -1.0007E+01
       6       8     132     Tout - Tin OF BED1                 5.0782E+02   -1.0302E-01    -1.0302E+01
       7       9     134     phi=(Tout-Tin)/Tin OF BED1         4.0173E+00   -9.0876E-04    -9.0876E-02
       8      17       9     Tgas at BED2 OUTLET                4.4112E+03    5.8337E+00     5.8337E+02
```

```
        9        18       132    Tout - Tin OF BED2             4.2847E+03      5.8308E+00      5.8308E+02
       10        19       134    phi=(Tout-Tin)/Tin OF BED2     3.3896E+01      4.5337E-02      4.5337E+00

><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><

  *** TIME (s) =     90.000         SYSTEM TIME STEP #    9000   ***

  FUNCTION-CONTROLLED VARIABLES FOR HEAT TRANSFER:
    COUNTER   ID #    ............NAME............      ARGUMENT F #       VALUE

  *** TIME (s) =     90.000         SYSTEM TIME STEP #    9000   ***

  FUNCTIONS FOR HEAT TRANSFER:
    COUNTER     F #     ID #     ............NAME............        VALUE        new-old VALUE  TIME DERIVATIVE

><><><><><><><><><><><><><><><><><><><><><><><><><><><><<. ><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><

  *** TIME (s) =     90.000         SYSTEM TIME STEP #    9000   ***

  FUNCTION-CONTROLLED VARIABLES FOR THE SYSTEM:
    COUNTER   ID #    ............NAME............      ARGUMENT F #       VALUE
       1       35     MASS FLOW RATE BC AT NODE 1           10           3.5900E-04

  *** TIME (s) =     90.000         SYSTEM TIME STEP #    9000   ***

  FUNCTIONS FOR THE SYSTEM:
    COUNTER     F #     ID #     ............NAME............        VALUE        new-old VALUE  TIME DERIVATIVE
       11        5        1      PROBLEM TIME                     9.0000E+01      1.0000E-02      1.0000E+00
       12       10      114      TABLE, FLOWRATE BC VS TIME       3.5900E-04      0.0000E+00      0.0000E+00
```

End of Sample Problem

Index

Notes

notes

notes

Notes

1 Aldemir, Tunc
Department of Mechanical Engineering
Ohio State University
206 West 18th Avenue
Columbus, OH 43210

1 Anghaie, Samin
University of Florida
202 Nuclear Sciences Center
Gainsville, FL 32611

1 Barattino, Maj. William
Phillips Laboratory/TAS
Kirtland AFB, NM 87117-6008

1 Bennett, Gary L.
NASA Headquarters
Code RP
Washington, DC 20546

1 Best, Frederick R.
Texas A&M University
Nuclear Engineering Department
129 Zachry Engineering Ctr.
College Station, TX 77843-3133

1 Bhattacharyya, Samit K.
Argonne National Laboratory
9700 South Cass-Bldg. 207
Argonne, IL 60439-4841

1 Bleeker, Lt. Col. Gary
3550 Aberdeen Avenue SE
Phillips Laboratory/VT
Kirtland AFB, NM 87117-5776

1 Borowski, Stanley K.
NASA Lewis Research Center
Adv Space Analysis Office
21000 Brookpark Road
Mail Stop AAC-2
Cleveland, OH 44135

1 Breen, J. A.
Pratt & Whitney GESP
P. O. Box 109600
West Palm Beach, FL
33410-9600

1 Brengle, Bob
Rockwell International Corporation
Rocketdyne Division
6633 Canoga Avenue
Canoga Park, CA 91303

1 Buden, David
Center for Nuclear Engineering
and Technology
INEL
P. O. Box 1625
Idaho Falls, ID 83415-2516

1 Buska, John
Mail Stop K551
P. O. Box 1663
Los Alamos National Laboratory
Los Alamos, NM 87545

1 Carroll, Wade P.
U. S. Department of Energy
NE-52
19901 Germantown Road
Germantown, MD 20874

1 Clark, John
NASA Lewis Research Center
21000 Brookpark Rd.
Cleveland, OH 44135

1 Cappiello, Mike
Los Alamos National Laboratory
MS K551
P. O. Box 1663
Los Alamos, NM 87545

1 Difilippo, Felix C.
Oak Ridge National Laboratory
P.O. Box 2008, Bldg. 6025, MS-6363
Oak Ridge, TN 37831-6363

1 Donahue, Benjamin B.
Boeing Aerospace
P. O. Box 240002
MS JX-23
Huntsville, AL 35824-6402

2 El-Genk, Mohammed
Institute for Space Nuclear
Power Studies
Chemical and Nuclear Engineering
Department
University of New Mexico
Albuquerque, NM 87131

1 Emrich, Bill
MS EP53 Flash 4666
NASA Marshall Space
Flight Center
Huntsville, AL 35812

1 Felty, Jim
U. S. Dept. of Energy
DP-241
19901 Germantown Rd.
Germantown, MD 20585

1 Gerrish, Harold
NASA Marshall Space
Flight Center
Huntsville, AL 35812

1 Glass, James F.
Systems Engineering & Analysis
Rocketdyne Division
Rockwell International Corp.
6633 Canoga Avenue
Canoga Park, CA 91303

1 Haloulakos, V. E.
McDonnell Douglas Space Sys Co
Internal Mail Code A3-Y833-13/3
5301 Bolsa Avenue
Huntington Beach, CA 92647

1 Haman, Nelson A.
Argonne National Laboratory
9700 So. Cass Ave.
Building 207
Argonne, IL 60439-4841

2 Hampston, Ken
3550 Aberdeen Avenue SE
Phillips Laboratory/VT-X
Kirtland AFB, NM 87117-5776

1 Haskin, Eric F.
Institute for Space Nuclear
Power Studies
Chemical and Nuclear Engineering
Department
University of New Mexico
Albuquerque, NM 87131

1 Haslett, Robert
Grumman Aerospace Corp.
MS B09-25
Bethpage, NY 11714

1 Hendrix, Robert
NASA Lewis Research Center
Techpark 1
Space Propulsion Technology Div.
Cleveland, OH 44135

1 Hodgdon, Maj. Ken
3550 Aberdeen Avenue SE
Phillips Laboratory/VT-PN
Kirtland AFB, NM 87117-5776

1 Howe, Steve
Los Alamos National Laboratory
P. O. Box 1663
MS F643
Los Alamos, NM 87545

1 Husser, Dwayne
Babcock & Wilcox
P. O. Box 11165
Lynchburg, VA 24506-1165

1 Jones, Lee
MS EP53 Flash 4666
NASA Marshall Space
Flight Center, AL 35812

1 Kerrebrock, Jack L.
Massachusetts Institute of
Technology
Rm. 33-411
77 Massachusetts Ave.
Cambridge, MA 02139-4307

1 Klein, Andrew C.
Department of Nuclear Engineering
Oregon State University
Radiation Center A112
Corvallis, OR 97331-5902

1 Kuck, Inara
3550 Aberdeen Avenue SE
Phillips Laboratory/VT-P
Kirtland AFB, NM 87117-5776

1 Latham, T. S.
United Technologies Research Ctr.
Silver Lane
East Hartford, CT 06108

1 Lee, Stacy K.
Texas A&M University
Nuclear Engineering Department
129 Zachry Engineering Ctr.
College Station, TX 77843-3133

1 Ludewig, Hans
Brookhaven National Laboratory
Building 701
Upton, NY 11719

1 Luth, W. C.
DOE/ER
U. S. Dept. of Energy
ER-15, J317/GTN
19901 Germantown Rd.
Germantown, MD 20585

1 Maise, George
Brookhaven National Laboratory
Bldg. 701
Upton, NY 11973

1 Mangus, Jim
Westinghouse Electric Corporation
Box 158
Madison, PA 15663-0158

1 Martinell, John A.
Idaho National Engineering
Laboratory
P. O. Box 1625
Idaho Falls, ID 83415

1 Matson, Warren
Idaho National Engineering
Laboratory
P. O. Box 1625
Idaho Falls, ID 83415

1 McElroy, Steve H.
Babcock & Wilcox
P. O. Box 11165
Lynchburg, VA 24506-1165

1 McDaniel, Patrick J.
Phillips Laboratory/WSA
Kirtland AFB, NM 87117-6008

2 Metzger, John D.
Grumman Electronics Systems
Division
MS B09-25
Bethpage, NY 11714

1 Miller, Tom
NASA Lewis Research Center
21000 Brookpark Rd.
Cleveland, OH 44135

1 Mims, James E.
Advanced Systems Analysis
6739 Academy Road NE
Albuquerque, NM 87109-3345

1 Moore, R. L.
Advanced Reactor Design
EG&G, Inc.
2145 E. 17th Street
Box 1625
Idaho Falls, ID 83415-1575

1 Parsley, Randy C.
Rocket Conceptual Design
Pratt & Whitney
P. O. Box 109600
Mail Stop 714-65
West Palm Beach, FL 33410-9600

1 Pelaccio, Dennis G.
Institute for Space Nuclear
Power Studies
University of New Mexico
Albuquerque, NM 871311

1 Perkins, Ken
Brookhaven National Laboratory
Bldg. 701
Upton, NY 11973

1 Powell, James R.
Building 701
Brookhaven National Laboratory
Upton, NY 11973

1 Reid, Hank
Babcock & Wilcox
Space Systems Engineering
Mt. Athos Road - Rt. 726
P. O. Box 11165
Lynchburg, VA 24506-1165

1 Roberds, Richard M.
Engr. Science & Math
University of Tennesee Space
Institute
B. H. Goethert Parkway
Tullahoma, TN 37388

1 Sapyta, Joseph J.
Babcock & Wilcox
Space Systems Engineering
Mt. Athos Road - Rt. 726
P. O. Box 11165 - MC#08
Lynchburg, VA 24506-1165

1 Schuller, Michael J.
Phillips Laboratory/AWYS
Kirtland AFB, NM 87117-6008

1 Shepard, Kyle
General Dynamics Space
Systems Division
P. O. Box 85900
San Diego, CA 92138

1 Sheppard, Susan J.
Babcock & Wilcox
Space Systems Engineering
Mt. Athos Road - Rt. 726
P. O. Box 11165 - MC#08
Lynchburg, VA 24506-1165

1 Smith, John M.
NASA Lewis Research Center
MS 301-5
21000 Brookpark Rd.
Cleveland, OH 44135

1 Solon, Marty
Grumman Space Electronics
Division
South Oyster Bay Road
MS B09-25
Bethpage, NY 11714

1 Suzuki, David E.
3550 Aberdeen Avenue SE
Phillips Laboratory/VT-X
Kirtland AFB, NM 87117-5776

1 Tepes, Frank
Grumman Space Electronics
Division
MS B0-25
Bethpage, NY 11714

1 Walter, Carl E.
Lawrence Livermore
National Laboratory
7000 East Avenue L-144
MS L-197; P. O. Box 808
Livermore, CA 94550

1 Walter, P. B.
Nuclear Engineering Department
Pennsylvania State University
231 Sackett Building
University Park, PA 16802

1 Walton, Lewis A.
Babcock & Wilcox
3315 Old Forest Rd.
Lynchburg, VA 24506

1 Warren, John
Division of Defense Energy
Projects
Office of Nuclear Energy
U. S. Dept. of Energy, NE-52
Washington, DC 20545

1 Watson, Ron
Babcock & Wilcox
Space Systems Engineering
Mt. Athos Road - Rt. 726
P. O. Box 11165
Lynchburg, VA 24506-1165

1 Zubrin, Robert
Martin Marietta Astronautics
P. O. Box 179
Denver, CO 80201

1	8523-2	Central Technical Files
5	7141	Technical Library
1	7151	Technical Publications
10	7613-2	Document Processing for DOE/OSTI
	332	M. E. Vernon
	1511	F. G. Blottner
	1511	C. E. Hickox
	1511	C .C. Wong
	1512	R. J. Gross
	1513	R. L. Akau
	1513	R. E. Hogan
	1513	R. D. Skocypec
	1672	L. R. Shipers
	2723	D. F. Beck
	4115	L. W. Connell
	6216	D. R. Gallup
	6342	R. S. Longenbaugh
	6342	L. C. Sanchez
	6400	N. R. Ortiz
	6404	D. A. Powers
	6412	V. J. Dandini
	6412	S. E. Dingman
	6412	G. W. Wyss
	6413	F. T. Harper
	6414	J. E. Kelly
	6418	S. L. Thompson
	6422	J. E. Brockmann
	6429	K. E. Washington
	6474	L. O. Cropp
	6474	J. R. Hipp
2	6474	K. S. Kwok
	6474	J. H. Lee
	6474	A. C. Marshall
	6474	F. J. Wyant
	6500	J. K. Rice
	6501	J. V. Walker
25	6513	D. Dobranich
2	6513	C. D. Harmon
	6513	D. L. Miller
	6513	C. A. Ottinger
	6513	D. S. Oscar
	6513	G. F. Polansky
	6513	G. S. Rightley
	6514	S. C. Bourcier
	6514	R. L. Coats
	6514	E. J. Parma
	6514	P. S. Pickard
	6514	A. J. Suo-Anttila
	6514	D. G. Talley
	6514	S. A. Wright
	6515	M. Berman
	6515	K. W. Boyack
	6515	W. H. McCulloch
	6515	M. J. Rightley
	6515	M. P. Sherman
	6515	M. F. Young
	6521	J. W. Bryson
	6505	J. S. Philbin
	6523	K. R. Boldt
	6607	G. C. Allen

⭑ U.S. GOVERNMENT PRINTING OFFICE 1993—573-110/80282

DATE
FILMED

2 / 15 / 94

END